



Web Based Restaurant Management System

**A dissertation submitted for the Degree of Master of
Information Technology**

W.B.A.C. Piyatissa

University of Colombo School of Computing

2020



Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: W. B. A. C. Piyatissa

Registration Number: 2017/MIT/055

Index Number: 17550552

Signature:

Date:

This is to certify that this thesis is based on the work of under my supervision.
The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: Dr. H.A. Caldera

Signature:

Date:

Dedication

To my parents.

The reason of what I become today. Thank
you for your great support and continuous
care.

Acknowledgement

First I would like to express my sincere gratitude to Dr. H. A. Caldera under whose supervision this project was undertaken, for his guidance and persistent encouragement. His valuable advices and criticisms drive this project more successful.

Secondly I wish also to express my appreciation to my client Mr. Isuru Ranawaka and his staff for the guidance, support and encouragement given through the journey.

I express my heartiest gratitude to my father Mr. W.B. Piyatissa, my mother Mrs. Padma Chandrathilaka who were with me during the project period by providing advice, guidance and valuable support to make everything success. Appreciation is also extended to my friends Mrs. Tharini Kananke and Mr. Damith Kandage for their Friendly corporation and support during this project.

Last, but not least. I would like to express my grateful acknowledgement to University of Colombo School of Computing for offering this Masters of Information Technology Degree program to individuals like myself who is willing to gain knowledge in IT sector, and also all the staff members who guide student from the beginning of the program.

Abstract

Matara Kema is a restaurant which located near Godagama expressway interchange in Matara. Currently they are offering their services to large number of customers daily. Their customer base is growing day by day because of their location and quality of their food. Even though this is a small restaurant, the service they are offering is huge.

With the growing customer base, it is difficult to offer services to all of them manually. Therefore, this restaurant requires to move to a computerized solution. This will help to run their operations smoothly and in more efficient manner.

Suggested system allows Matara Kema restaurant to increase the scope of the business with the little staff they have. As this is a growing business which is in its first stage, they still cannot go for more staff. Current trend is also going for an online solution. This system allows restaurant to quickly and easily manage online menus and customers can navigate and place orders really fast. Restaurant staff also quickly view orders and prepare orders without previous miscommunication. Graphical interfaces are really efficient and effective for both customers and restaurant staff

For the managers and owners this system is very effective because they can view their requirements in reporting module.

This system has used MVC (Model View Controller) architecture. Object Oriented approaches have been used in developing. For analysis and design, Uniform resource modeling was used. Php, html, JavaScript ect has used to develop the website. mySQL has been used to create to databases. Lanka hosting space was used to host the website.

Any web browser such as Edge, Chrome, Firefox can be used to access the website in any environment such as windows or Linux.

Table of Contents

Chapter 1	Introduction	1
1.1	Introduction.....	1
1.2	Overview.....	1
1.3	Problem	1
1.4	Motivation for the project	3
1.5	Objectives of the project	3
1.6	List of deliverables.....	4
1.7	Scope of the project	5
1.8	Outline of the chapters	6
1.9	Summary	7
Chapter 2	Background.....	8
2.1	Introduction.....	8
2.2	Overview.....	8
2.2.1	Benefits of Restaurant Management Systems	10
2.3	Review of Similar Systems	13
2.3.1	Subway Fast food restaurant.....	13
2.3.2	Pizza Hut.....	18
2.4	Comparison of development life cycles.....	20
2.5	Justification of a single design strategy for implementation	21
2.6	Summary	21
Chapter 3	Methodology.....	22
3.1	Introduction.....	22
3.2	Analysis.....	22
3.2.1	Nature of the current method.....	23
3.2.2	Requirements of the proposed system	25
3.2.3	Non-functional requirements	27
3.2.4	Proposed architecture for the system.....	27

3.3	Design	28
3.3.1	Design Techniques	28
3.3.2	User Interface Design	32
3.4	Summary	33
Chapter 4	Implementation	34
4.1	Introduction.....	34
4.2	Implementation Environment	34
4.2.1	Operating Environment	34
4.3	Development tools and techniques	34
4.3.1	Tools	34
4.3.2	Technologies.....	35
4.4	Reusable Components.....	35
4.5	Network Architecture.....	35
4.6	Application Development Architecture	35
4.7	Main User Interfaces.....	36
4.7.1	Admin Dashboard.....	36
4.7.2	Order Food Cashier	34
4.7.3	Kitchen Module	34
4.8	Summary	35
Chapter 5	User Evaluation and Testing.....	36
5.1	Introduction.....	36
5.2	Testing Procedure	36
5.3	Test Plan and Test Cases.....	36
5.3.1	Master Data modules	37
5.4	Acceptance Testing	38
5.5	Summary	38
Chapter 6	Conclusion	39
6.1	Introduction.....	39

6.2	Problems Encountered	39
6.3	Lessons Learnt	39
6.4	Future enhancements	40
Appendix A	System Design	43

List of Figures

Figure 2.1 : Top Features Desired by Restaurant Owners/Managers.....	9
Figure 2.2 Restaurant Management system example	12
Figure 2.3 Subway-Front view of the website.....	13
Figure 2.4 : Subway-Navigation Menu	14
Figure 2.5: Subway-Full menu Part – 1.....	14
Figure 2.6: Subway-Full menu Part – 2.....	15
Figure 2.7: Subway-Start Ordering	15
Figure 2.8: Subway-Order by clicking on food item in menu.....	16
Figure 2.9: Subway- Order Online	16
Figure 2.10: Subway-Nutrition details	17
Figure 2.11: Subway- Other Items	17
Figure 2.12: Pizza hut- Front View	18
Figure 2.13: Pizza hut- Select outlet.....	18
Figure 2.14: Pizza hut- Menu	19
Figure 2.15: Comparison of software development life cycles.....	20
Figure 3.1: Current Restaurant Management	23
Figure 3.2 : High level Use case diagram representing the existing system.	24
Figure 3.3: MVC framework.....	27
Figure 3.4: Overall High-level Use case for the proposed system	29
Figure 3.5: Class diagram.....	30
Figure 3.6: Activity diagram- Customer order	31
Figure 3.7: User Privileges	32
Figure 4.1: Admin Dashboard	36
Figure 4.2: Cashier Order Module.....	34
Figure 4.3: Kitchen Order Ticket	34
Figure 4.4: Home Page of the system.....	35
Figure 4.5: Food Menu	35
Figure A.1: Customer Module.....	43
Figure A.2 level diagram.....	44
Figure A.3 : level 1 diagram.....	45
Figure A.4: Level 2 diagram.....	45

Figure A.5: Level 2 diagram.....	46
Figure A.6: Level 2 diagram-2	46
Figure A.7 : Menu Pricing	49
Figure A.8: Sub Menu	49
Figure A.9: Selecting food item	49

List of Tables

Table 3-1: Required functionalities to the stakeholders	25
Table 5-2: Add customer test case.....	37
Table A.1 Customer Order Food Online	43
Table A.2 : Level 0 diagram.....	44
Table A.3 : Category Details	47
Table A.4 : FoodItems Details.....	47
Table A.5: Item Master Table.....	47
Table A.6: Table Master	48
Table A.7: Order Details Table	48

List of abbreviation

MIT	Master of Information Technology
WBRMS	Web Based Restaurant Management System

Chapter 1 Introduction

1.1 Introduction

Overview of the problem is described in this chapter, with highlighting the importance of the problem. In the next sub section, the motivation is outlined. Then the objectives of the project, list of deliverables and scope of the project are explained in the next sub sections respectively. Finally, the chapter outline is summarized.

1.2 Overview

“Matara Kema” is a restaurant located in Matara. They are handling food ordering process and daily routine of their restaurant manually. Customers cannot order food online in this hotel. In case of take away food or dining there, customer have to visit or call and order. Table reservation is also the same. This project is based on automating the above mentioned processes.

The problem that many businesses face today is to make sure that they attract new customers and also they keep their existing customers.

The cost to attract a new customer is costlier than retaining the old customer. Therefore, there is an argument that for a business, existing customer is worthier than a new customer. In this industry, a customer is likely to return to the restaurant in the future if they received an excellent customer service as well as appetizing food [1]. However, if they had to wait for an unreasonable amount of time or there was a mistake in the order, it’s very unlikely the customer would return.

1.3 Problem

“Matara Kema” is a food store located in Matara. Their process is currently manual and time consuming. Customer cannot order food online, before they visit the restaurant. They have to visit the restaurant and place the order. Then they have to wait until the order is ready.

There is no visual confirmation that the order was placed correctly. Customers cannot ensure about the correctness of their order. Order customization is also difficult because it takes more time. Table reservation is also difficult.

This restaurant is very busy in the morning and evening after 6 pm because the restaurant is located in a very attractive place, near Godagama expressway interchange. Therefore, the restaurant is always full. Currently customers have to come to the restaurant to reserve a table. Sometimes customers have to wait long time to reserve a table during busy times. Also, very long queues are there. It is difficult for the cashier to handle such a situation. On the other hand,

this situation is difficult for the kitchen also. They are receiving orders one by one. Sometimes kitchen is receiving same food item within separate orders in same time. So kitchen has to make them separately. This is time consuming and inefficient costly method. It is better if the orders for same kind of food are grouped. It will help them to save time and attend to the next order quickly.

At the end of the day, taking a summary is also difficult because there are so many orders and they are not recorded properly. This happens because they are not keep records properly. Taking this summary is important because they can review what the fast moving items are and what items are not ordered frequently. Sometimes customers do valuable suggestions. Usually staff is listening to them and try to do the improvements. But they are not keeping records of those suggestions and they may forget them at the end of the day. It is not good for the restaurant because it will break the trust of the customer and missing the opportunity for the improvement.

Currently, this restaurant uses least efficient methods such as paper-based or verbal method to communicate between the restaurant and kitchen. Even though this approach is implemented in successful profitable restaurants, there are several problems which could be seen as reducing the restaurant's efficiency that can be identified using above scenario:

- Sometimes handwriting can lead to miscommunication.
- Order logging is unmanageable.
- Inefficient restaurant-kitchen communication.
- Difficult order tracking and time management.
- Difficult stock management.
- Limited statistical output [2].

Online system that proposed here will simplify the ordering process and it will be helpful to both restaurant and customer. The proposed system will be developed with interactive menus, pop-up messages etc. for the easiness of the user. Customer can select the required food item with available customizations and the can change the order at any time before checking out online. If order is confirmed, it will display a pop-up message as confirmation to the customer. When the order is placed it's recorded in the database and retrieved in real time manner. Through this option Restaurant Employees are allowed to quickly go through the orders which they have received. Therefore, they can process all orders efficiently and effectively with minimal delays and confusion.

1.4 Motivation for the project

The motivation of this project comes with observing their difficulties in busy situation while I was there as I usually visit the place to order food. Personally I don't have much time to wait in long queues. This restaurant is also very crowded during lunch and dinner time. Although this restaurant is not very large they have to provide quality service to enormous number of customers. Without a system it is very difficult. Other than that, I value learning web designing and development because I have less experience in this area and it will be helpful in the future for my career.

New expectations are there for this project due to the current situation in the country with Covid-19 virus. This kind of solution will help to make the restaurant less crowded.

1.5 Objectives of the project

Maximizing the profit is one of the main objectives of any business. This can be achieved by increasing efficiency and decreasing overheads without compromising customer satisfaction. Through better application of daily operations, a restaurant can increase its efficiency and can offer improved services to the customers. Because almost all processes are manual and time-consuming, all the processes should be automated.

The Main Objective: To build a web-based restaurant management system for "Matara Kema" Restaurant.

In order to fulfill the main objective, the following goals have to be achieved.

- **Improve customer relationship management**

The proposed system enables visual confirmation to the customers that the order was placed correctly and will decrease difficulties. When the order is ready, the kitchen can update the food order status as ready. At the same time, the customer and the cashier will be notified. This will reduce miscommunication and workload of the cashier. Reducing waiting time of the customer will improve the customer satisfaction.

- **Avoid long queues**

This solution will help to increase the efficiency of the restaurant's staff. It eliminates paper work and increases the level of accuracy. Staff can handle more customers in little time because a web-based solution can improve the speed of service, sales volume, and customer satisfaction.

- **Bulk Processing**

Order retrieval is simple and kitchen can see the order as bulks. Then kitchen can process more orders because they can complete several same type of orders at the same time.

- **Customer feedback**

Customers can give feedback which is very valuable for improvements of the restaurant.

- **Stock Control**

All the kitchen ingredient stock levels can be maintained through the system. Proposed system will facilitate restaurant to maintain kitchen stock.

- **Menu item management.**

Kitchen can maintain possible meals and can update which item cannot be provided in relevant day.

- **Discounts**

Providing special discounts and promotions. They can assign, "Item of the Day" for special discount. System can identify whether the customer is new customer or regular customer. For the regular customer, system can provide special discounts.

- Increase the customer satisfaction and make them retain with the restaurant.

1.6 List of deliverables

The following deliverables are expected upon the completion of the proposed System.

- Create WBRMS where the customers can order the items, update and cancel orders, staff can manage menus, discounts, display items and control inventory.
- Generate essential reports throughout the process.
- Provide detailed user manual.

1.7 Scope of the project

Proposed system is valuable for both customer and the restaurant because it is simplifying the order processing process.

Customers have to create an account with valid phone number or email and can log-in to the system. The web page has up-to-date and interactive menu with all the available food items. When customer made a selection, items added to their order. Customer can review order at any time and change the selection before the payment. Online payment and cash on the collecting counter is possible. Confirmation is prompt to the customer. If it is required, customers can check the table availability and make a table reservation if necessary. Proposed system is not handling delivering food, due to restaurant is not having delivery team.

After a customer placed an order, order details are visible to the kitchen. They can see what the quantity that is required from each item is and they can fulfill multiple orders with same food item in same time. This happens because proposed system can combine orders during allowed time period.

Stock of the ingredients should be always up to date. Otherwise, restaurant cannot fulfill the customer orders properly and may accept orders that cannot be fulfilled due to lack of ingredients. Real-time view of ingredient stock levels is very much important to so much necessary to any restaurant. Then only the meals with enough ingredient stock can be sold. The stock levels will be updated by the kitchen staff at the end of the day through the proposed system.

Restaurant should be able to control the menu items. This ensures that the customers can only order available food items. Only authorized employee can handle the menu. They can create and remove food categories, food items. When creating these categories and food items, employee can add photos and description also. Adding new food items also possible. Remove food items and change visibility of food items. Create and remove options of food items. They can edit and update prices of the food items. They can select “Food Item of the day” and allocate special discount to them.

Giving reasonable discounts to the customers, can keep customers without moving out from the restaurants customer base. First restaurant must identify if the customer is new customer or regular customer. Giving proper discounts to regular customers will increase their satisfaction and they will not move to another restaurant.

Restaurant can keep track of all the orders, retrieve and display order information. This information is visible to the kitchen for the order processing. These order details can be useful to report generation.

Customers can give general feedback and suggestions about the restaurant and meals. Those are only visible to the management of the restaurant. Also, customers can give star rating to each and every meal. Other customer can view those when they are selecting their meals. Order logging and report generation are also implemented in proposed system.

1.8 Outline of the chapters

The dissertation offers overall clarification about the WBRMS for “Matara Kema” restaurant. Outline of the chapters is shown below.

Chapter 2 – Background

Second chapter describes the background of the system. This will describe similar systems that are available and alternative technologies available relevant to the system. It will include essential background information with references and strengths and weaknesses of those alternatives.

Chapter 3 - Methodology

Analysis and Designing of the system described here. Description about the current system, user requirements as well as functional and non-functional areas will describe here. It will describe the details about how system is designed. It will contain relevant UML diagrams etc.

Chapter 4- Implementation

Implementation plan of the system will describe in this chapter. Plan of implementing this system and what have been achieved up to date through results that already have accomplished will describe here. Furthermore, this chapter covers content of the changed work from the original proposal with reasons. And problem faced when implementing the system, the scope of the project compared with the time plan as well as result that have achieved. Furthermore, it will describe test plan that was used to verify and validate system, and description of the effect of errors.

Chapter 5 – Testing and evaluation

This chapter describes the testing techniques, test plans and client evaluation.

Chapter 5 –Conclusion

Experience gained through the project along with areas for future recommendations are discussed in this chapter.

1.9 Summary

This chapter has given the introduction to the project and to the documentation. Next chapter will give background study before analyzing the current situation.

Chapter 2 Background

2.1 Introduction

This chapter gives an insight to Restaurant Management systems similar in nature to that of the one being developed in this project. It also gives a brief introduction to the importance of requirement gathering, a discussion on the development methodologies available as well as a justification on the platform and software used in this project.

2.2 Overview

Use of Restaurant management system helps to manage staff, sales, customers in easier manner.

Restaurant Management helps to run a restaurant in more efficient manner. It helps to manage orders, menus, inventory and sales. This kind of system usually contains hardware, such a cash register, large displays or touch panels, barcode scanner and receipt printer as well as software. It provides a comprehensive tool that allows to see restaurant and its needs at a glance, which can simplify the workload on a day-to-day basis [3].

What is a restaurant management system?

“A restaurant management system (RMS) is a type of point-of-sale (POS) software specifically designed for restaurants, bars, food trucks and others in the food service industry. Unlike a POS system, and RMS encompasses all back-end needs, such as inventory to staff management. A system typically includes software and hardware, such as registers, scanners and receipt printers [3].”

Types of RMS currently available

Because every restaurant has unique needs, there are different RMS types to choose from. To determine which type best suits the restaurant, one has to focus on the features that business requires and how important each of those features [9].

1. End to end

This is the most robust and comprehensive type of RMS. Main features include core POS, inventory control, CRM, staff, menu, order and payment management, technical support, and reporting and analytics. Depending on the RMS vendor, developer may be able to mix and match features.

2. POS

This is the core of the system and allows to integrate it with third-party systems for inventory, accounting, marketing and other key systems.

3. iPad or Android only

Most systems are designed to run only on one device type to maintain the integrity of the system. Determine which device type will use in the restaurant.

4. General POS

This system is designed for businesses that have both retail and food services available. It offers seamless crossover with add-on modules [3].

Top Features Desired by Restaurant Owners/Managers

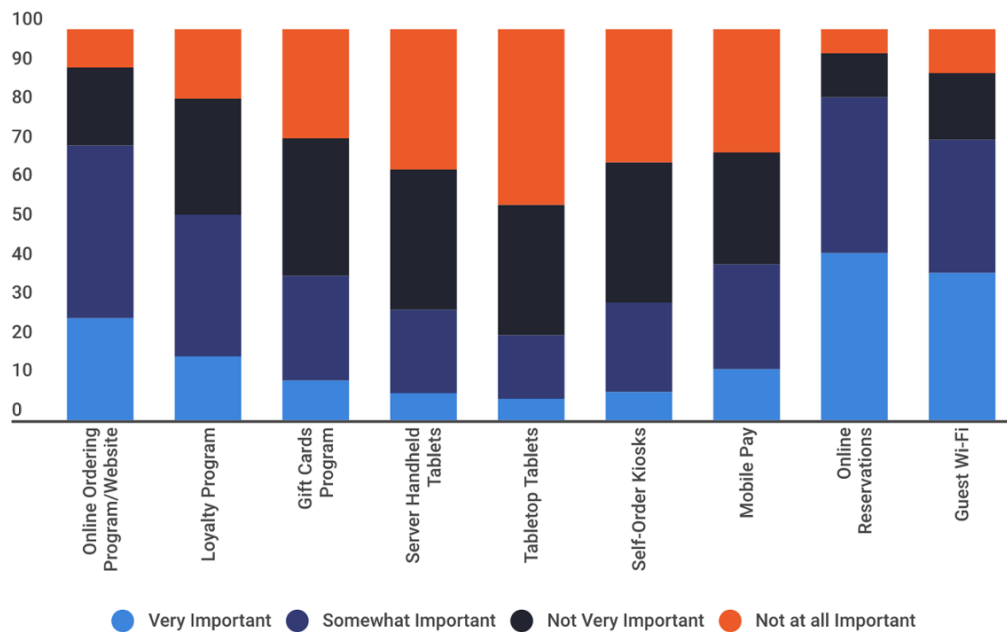


Figure 2.1 : Top Features Desired by Restaurant Owners/Managers

According to figure 2.1 online ordering and online reservations are the most required options that are required by restaurant owners and managers. Other than that loyalty and mobile pay is also required.

2.2.1 Benefits of Restaurant Management Systems

1.Track sales item by item

All transactions such as orders, payments, promotions, deals expenses are captured by the restaurant management system. Therefore, sales data is accurate and spot on to last item.

Therefore, revenues can be calculated correctly and health of the business operations can be identified accurately.

In built notifications are there to give notifications of variances of the system. Most of the restaurant management systems have this feature. As example, the gap between stock level and order volume at the end of the day, or unauthorized order voids. Alerts are typically sent via email or phone or can be viewed from a dashboard.

Sales data can also be broken down into various categories to make sense , such as by bestselling item ranking, customer names, or cash vs. card payment rate [4].

2. Generating quick and efficient financial statement

If it's possible to capture transactions digitally, manual errors can be avoided. As example, each transaction is time-stamped and recorded with relevant details, such as cashier name and sold item etc. Sales data is also synced with CRM and inventory to update customer information as well. The data is consolidated. With that integrity of the financial reports is well protected. Not like manual system, report generation is simple as few mouse-clicks. The system can run analytics on data queries. Using that KPIs can be displayed using graphs and charts.

3. Better customer service

This is probably the best reason to use a restaurant management system: make customers happy. Most restaurant management systems come with CRM software features. This module records customer information like name, contact details, and transactions. With more knowledge of customer's likes, delivering a more satisfying service is possible. This can build a mailing list, run campaigns that target their interest based on past orders, or push deals that match their profiles. The CRM typically comes with a loyalty program to help rewarding regular clientele.

Similarly, advanced features like online booking, mobile payments, and kiosk POS make it convenient for customers to place orders and transact business with restaurants. Similarly, the use of an iPad or Android tablet makes tableside ordering faster, to the delight of customers.

Some restaurant management systems also give the option to work with a shared delivery network service therefore can reach eager customers outside the neighborhood. Overall, the restaurant management system helps to run a restaurant business more efficiently to the satisfaction of the customers [4].

4. Data can be accessed from anywhere

Cloud-hosting method is used in most restaurants today. It allows to access the system and data from a web browser. Real-time sales transactions and tracking employee performance is allowed even on the go. That means manager is not required to stay in restaurant premises to handle the restaurant. Manager can be a hands-on manager while out of the place. Even for the client or investor this is beneficial because of the remote data access.

5. Staff management become more efficient

If employee scheduling is available in a restaurant management system, it will help to staff allocation during peak of off-peak time frames. Downtime may be less. With aggregation of sales data with staff schedule, it is possible to match demand with supply. It will help to ensure that resources are optimized, neither over- or under- utilized.

Other than that, most of the systems employee scheduling feature. Using this method, employees can plot their schedule where everybody sees the available and taken slots. Conflicts of schedule or nonappearance can be avoided using this feature. Waiting staff during peak hours can be managed using this. It is helpful because staff can manage their own schedule without any problem.

6. Better communication for the kitchen, wait staff, and cashier

The perennial problem in running restaurants is that somewhere along the way, someone mixes up information. The result: an angry customer whose order is, if not wrong, remains unaddressed or improperly billed. A restaurant management system does away with this scenario. When waiter takes orders via a handheld device, that data is automatically transmitted to the cashier, so the right transaction is captured and billing is accurate. At the same time, the handheld device transmits the order items to the kitchen. Barking orders to the head cook is a thing of the past [4].

7. Saving Costs

Well planned and managed employee schedule and well organized inventory restaurant can reduce the daily operational costs. Avoided variances or at least considerably reduced, can add

more savings from restaurant's profits. Likewise, there are plenty of areas where a well-managed restaurant can squeeze out savings.

Some vendors also hook the restaurants to an online marketplace and delivery service network, so they can scale operations minus the overheads that come with expansion. New restaurants grappling with marginal profits can cut down on losses by simply running their operations more efficiently [4].

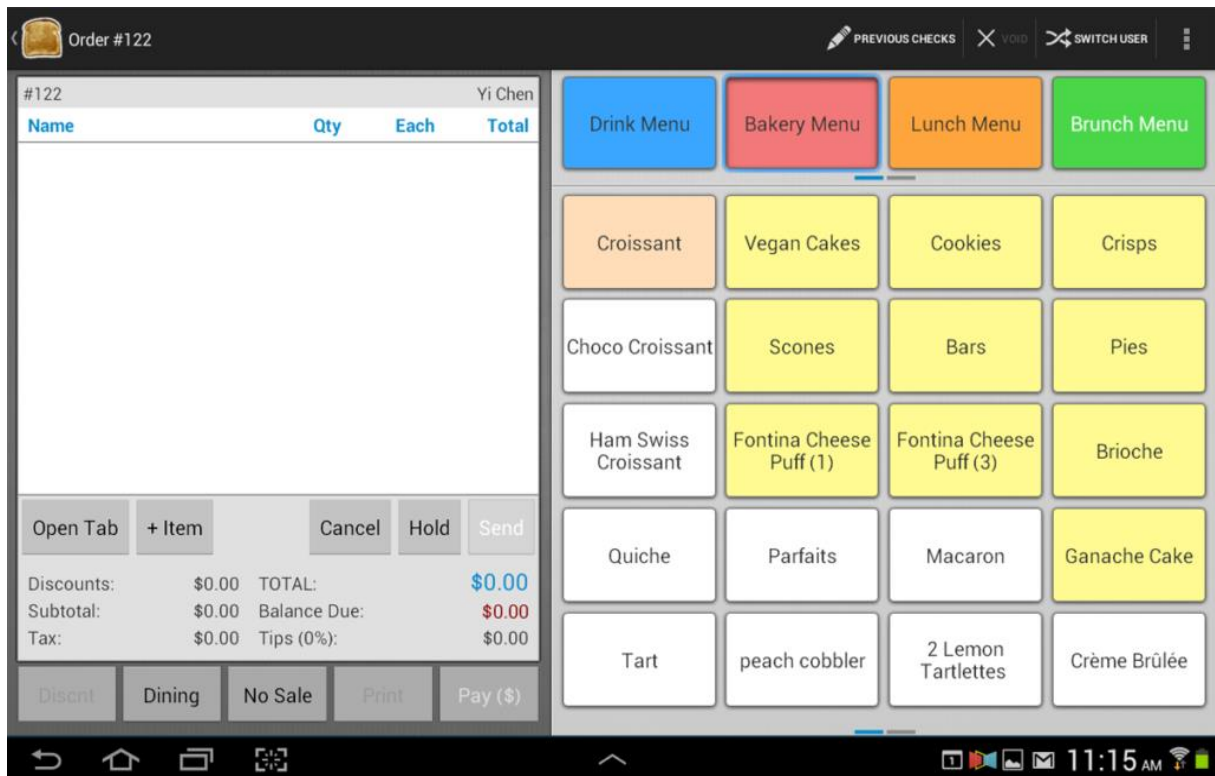


Figure 2.2 Restaurant Management system example

Given figure 2.2 shows an example for well-planned restaurant management system. Common ordering items are displaying in side section and current ordering items are visible clearly.

2.3 Review of Similar Systems

2.3.1 Subway Fast food restaurant

Subway is a nice website with good functionalities. It has two ordering options [11].

1. Order online
2. Order directly from restaurant

Online ordering

When customer is required to order food online can visit the website and first they can see daily deals. They can select from a deal if they wish.

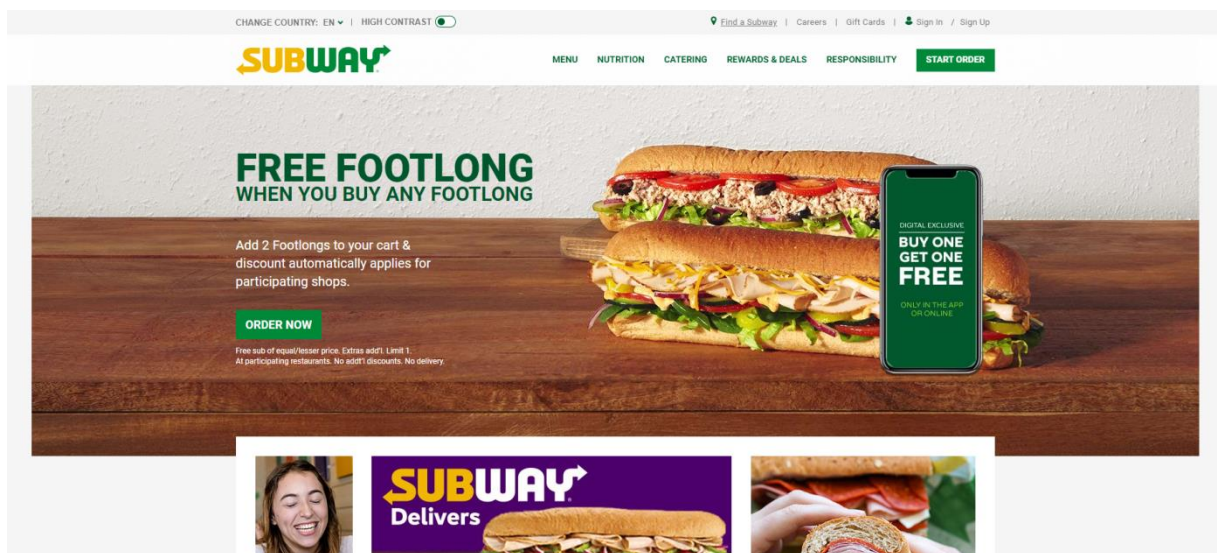


Figure 2.3 Subway-Front view of the website

Other than that customer can check the menus. In menu tab in front customer can go for selection or they can go for full menu.

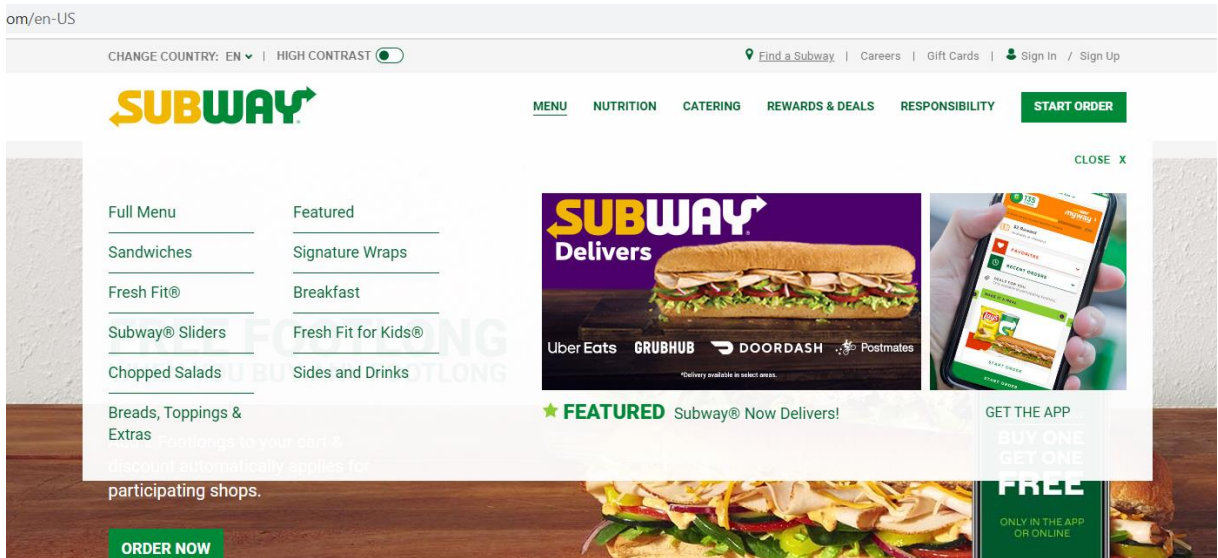


Figure 2.4 : Subway-Navigation Menu

The Menu is Highly organized and very clear. It is easier to select from these categories.

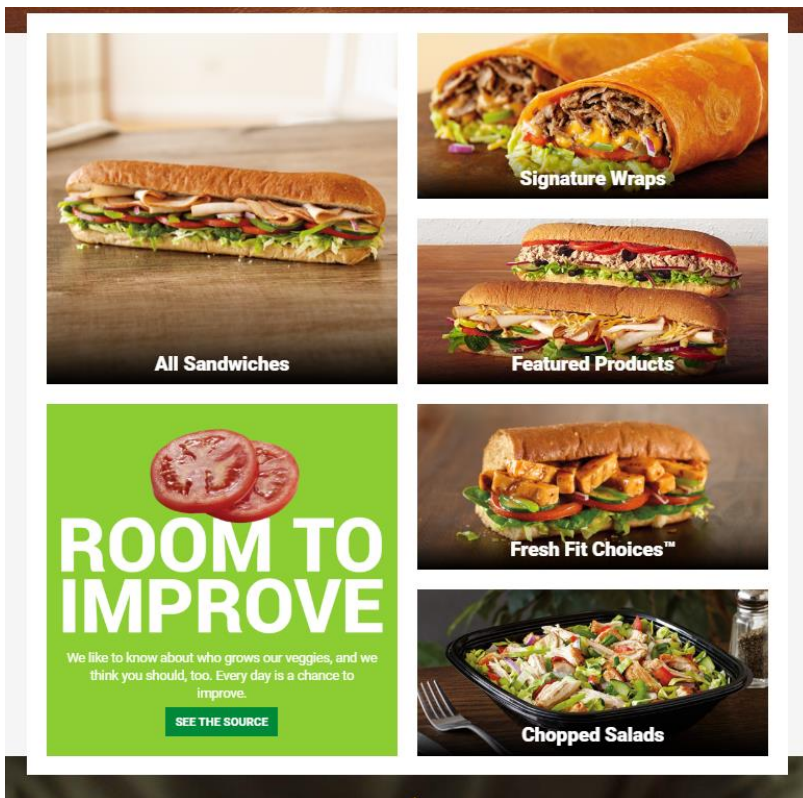


Figure 2.5: Subway-Full menu Part – 1

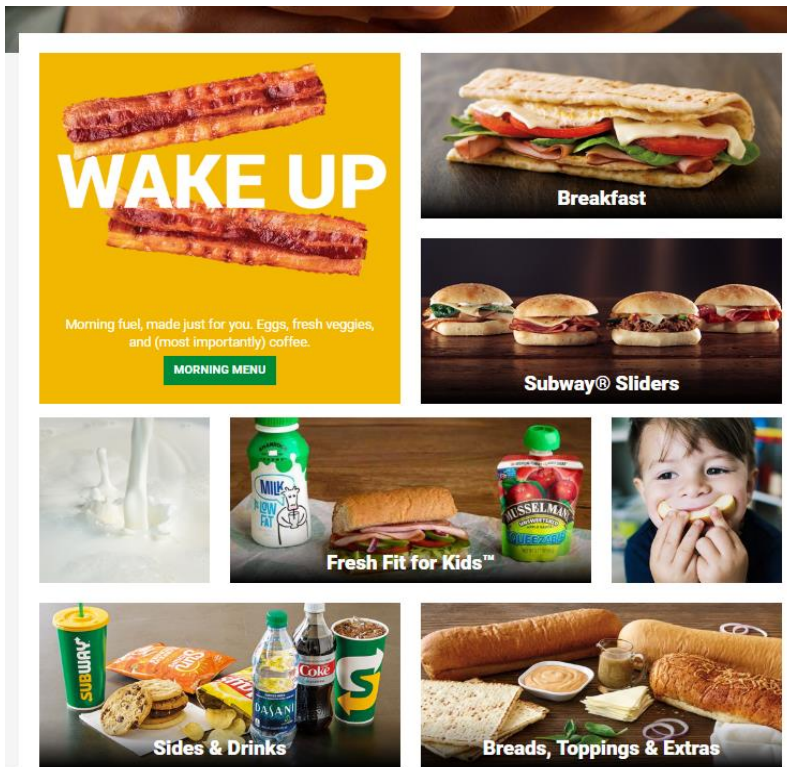


Figure 2.6: Subway-Full menu Part – 2

Ordering Process

When customer finish selection the can start ordering. Customer can directly do it by clicking food item or by clicking on “Start Order”

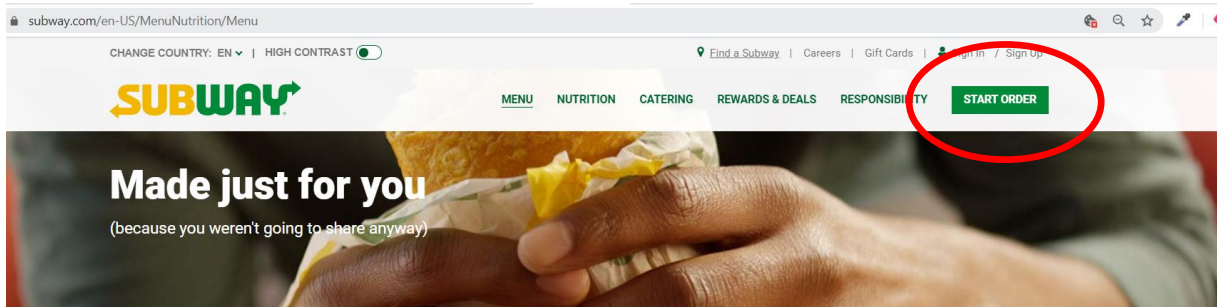


Figure 2.7: Subway-Start Ordering

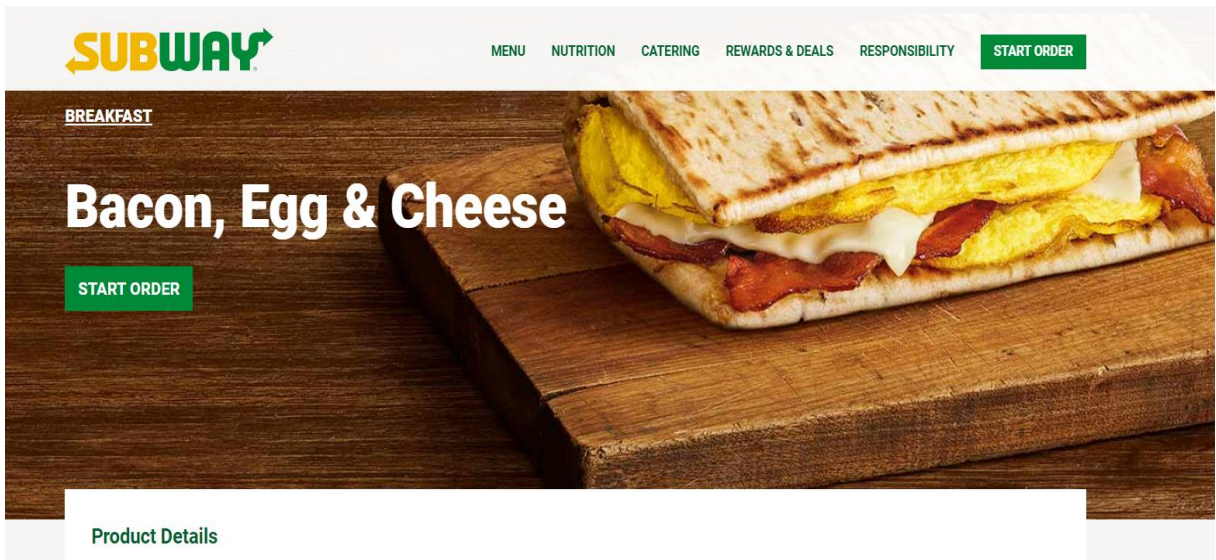


Figure 2.8: Subway-Order by clicking on food item in menu

When ordering customer has to fill the delivery details such as City, Mobile number, pick uptime and payment options ext.

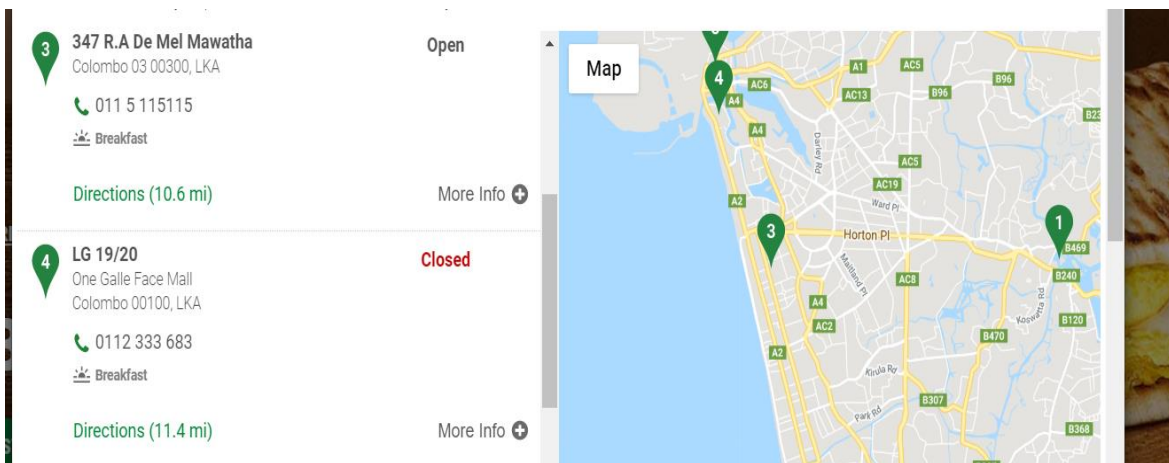


Figure 2.9: Subway- Order Online

Subway is letting their customers to use mobile application also. Their website has following useful functions [10].

- **Nutrition Information**

They provide nutrition information of food items.

Product Details

Start your day in a sizzlin' way with bacon, egg, and melty cheese on freshly toasted flatbread (or whatever you like). Pile on your favorite veggies and sauce. Start the day right.

Nutrition

198	450	20	7	0
SERVING SIZE (G)	CALORIES	TOTAL FAT (G)	SATURATED FAT (G)	TRANS FAT* (G)
190	1190	45	4	4
CHOLESTEROL (MG)	SODIUM (MG)	CARBOHYDRATES (G)	DIETARY FIBER (G)	SUGARS (G)
25	20	6	10	15
PROTEIN (G)	VITAMIN A % DV	VITAMIN C % DV	CALCIUM % DV	IRON % DV

NUTRITION CALCULATOR

Figure 2.10: Subway-Nutrition details

- **Responsibilities**

Company information, Plans etc. Is available here.

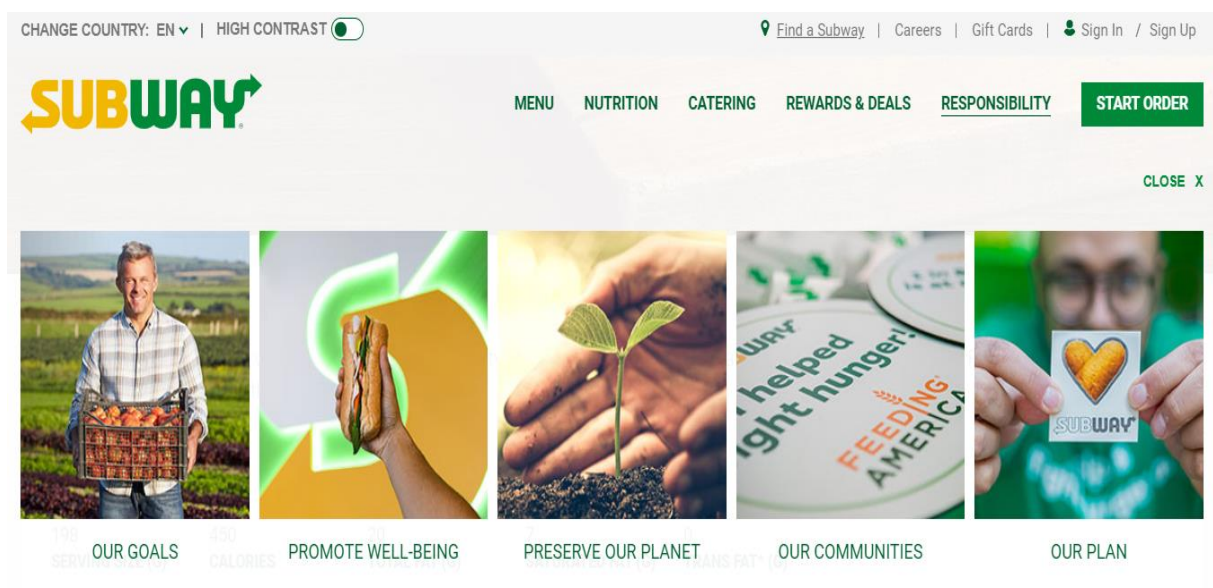


Figure 2.11: Subway- Other Items

Customers can register in this system free and do the online ordering. Other than that this website displays careers also. If customers wish they can send gift cards also. These are useful features of the Website. Website changes from location to location. But it is sad because some of these features are not available for Sri Lanka [11].

2.3.2 Pizza Hut

Pizza hut is a well known place in not only sri lanka but also all over the world. And this is populer than Subway and have more number of outlets [12].

Their website is also nice and attarctive. When comparing with subway –Sri Lanka, Pizzahut.lk is better.

This also has online ordering and Takeaway options

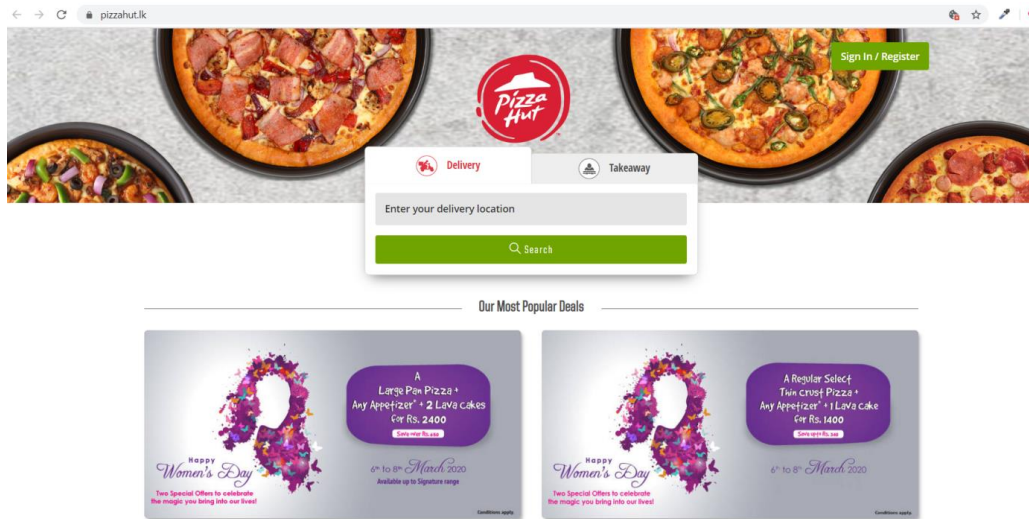


Figure 2.12: Pizza hut- Front View

First customer can select where to diliver or go for take-away. Before customer order to deliver they have to register on website. If customer wish to go for takeaway the can pre order in ther conviniant location.

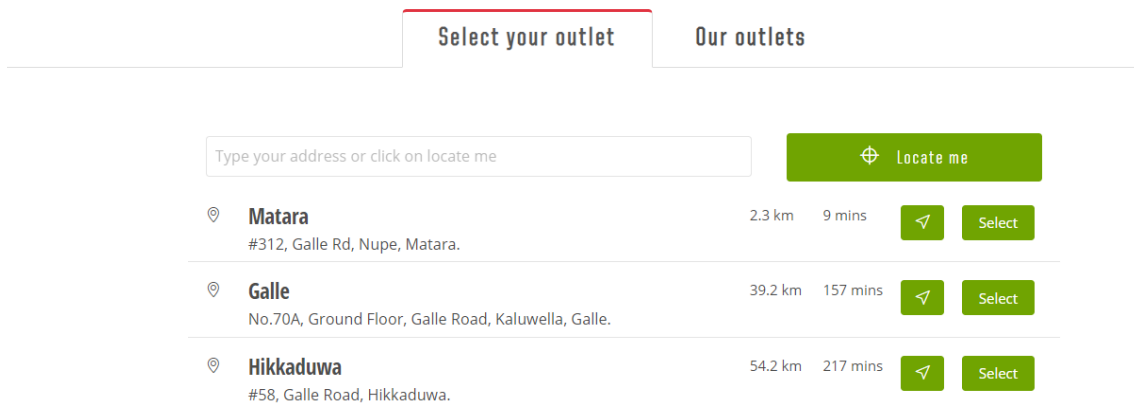


Figure 2.13: Pizza hut- Select outlet

Food menu is plain and simple but easy to order. With the menu customer can view the cart and know what is the total cost.

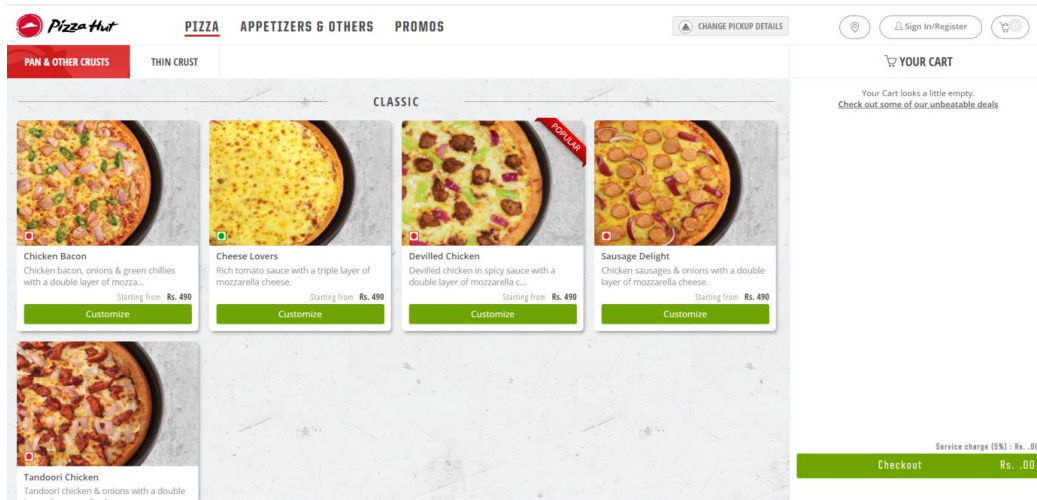


Figure 2.14: Pizza hut- Menu

Ordering pizza to deliver is basically contacting local outlet and it is very ease for both pizza hut staff and customer.

So many promotions are available and they can be different by the day of the week.

Navigation panel is not very good because this site works as simple functionality. Only ordering details are available. No extra details are available in this website.

By Considering above websites and current system in Matara-Kema resturent, following features and functionalities have been identified.

Useful Functions	Subway	Pizza Hut	Manual System	Suggested System
Role Base Access	✓	✓		✓
Menu Management	✓	✓	✓	✓
Nutrition Details	✓			✓
User Management	✓	✓		✓
Inventory Control			✓	✓
Order Management	✓	✓		✓
Offers and Discounts	✓	✓	✓	✓
Kitchen Management	✓	✓		✓
Purchase Management			✓	✓
Report Generation				✓

2.4 Comparison of development life cycles

It is very important to identify best method to do the development of the system. Variety of software development life cycle models are available. Success of any project is depending on selecting a suitable model for a specific development project. In the background study, waterfall, spiral, incremental/iterative (agile), v-model is considered as the alternative software development model.

Model/Feature	Waterfall	Spiral	Incremental/Iterative	V-Model
Specification of all the requirements in the beginning	Yes	Not all and frequently changed	Not all and frequently changed	Yes
Long term project	Inappropriate	Appropriate	Appropriate	Inappropriate
Complex project	Inappropriate	Appropriate	Appropriate	Inappropriate
Frequently changed requirements	Inappropriate	Appropriate	Appropriate	Inappropriate
Cost	Not costly	Costly	Costly	Costly
Cost estimation	Easy to estimate	Difficult	Difficult	Easy to estimate
Flexibility	Not	Less Flexible	Flexible	Little flexible
Simplicity	Simple	Intermediate	Intermediate	Intermediate
Supporting high risk projects	Inappropriate	Appropriate	Appropriate	Inappropriate
Guarantee of success	Less	High	High	High
Customer Involvement	Low	Low, after each iteration	High, after each iteration	Low
Testing	Late	At the end of each phase	After every iteration	develop test plan early on during the life cycle
Maintenance	Less maintainable	Maintainable	Maintainable	Little maintainable
Ease of Implementation	Easy	Complex	Easy	Easy

Figure 2.15: Comparison of software development life cycles.

Proposed Web based restaurant management system it is a middle sized project. Currently, requirements are finalized but some changes may possible. Therefore, project will be developed using a modular approach. The guarantee of the success of this project must be high.

Following a modular approach and develop the system as working modules will be better. Previously waterfall model was selected, but through further studies it confirmed that the traditional waterfall approach is not suitable for this kind of project. So, it is decided to follow the principles of iterative and incremental software development life cycle (Agile) for this project.

2.5 Justification of a single design strategy for implementation

MVC based CodeIgniter is a software approach that separate application login from presentation. In practice, it permits, web pages to contain minimal scripting since the presentation is separated from the Php scripting.

- The Model represents the data structure. Typically, model classes will contain functions that help you retrieve, insert and update information in the database
- The View is information that is being resented to a user. A view will normally be a web page, but in CodeIgniter, a view can also be a page fragment like a header or footer.
- The Controller serves as an intermediary between the Model, the View, and any other resources needed to process the HTTP request and generate a web page.

Object-oriented methodology is used by the CodeIgniter framework. This methodology offers a modular wise system development approach. Object-oriented approach combines data and processes into single entities called objects. This method has been utilized in developing the system as it makes the process easy by allowing reuse of components in coding

2.6 Summary

In this chapter similar kind of systems have been studied. Study about current situation and requirements also conducted. Next chapter will introduce the methodology of analyzing the system and designing system.

Chapter 3 Methodology

3.1 Introduction

This chapter is focused about stakeholders of the system, and requirements of the system. Also this chapter will look at the system design. System design is explained with the aid of diagrams to illustrate graphically certain sections of the software system.

Requirements gathering and analysis is a significant practice for a successful project. The main processes of this phase include domain understanding, requirements collection, classification, structuring, prioritization and validation. Appropriate methods and processes were engaged to carry out the analysis phase in an effective way.

Software design phase is an iterative process in which requirements gathered in analysis are translated into a “blueprint” in building the system. Afterwards this can be elaborated into detailed functional and behavioral requirements. The output of the design phase is the system specification.

3.2 Analysis

Existing method in this restaurant is all manual and study of the current system is done by using the following fact finding techniques.

Observations

Interviews

According to the information that gathered from interviews, employees have following problems.

Kitchen Staff and serving staff

It is difficult to remember large number of orders at once. And sometimes they have to prepare same food item several times. They prepare bulks. But it's difficult to pack the orders during busy times. If there is a method to view orders by food items(Bulks) it's easy to prepare. There should be a way to view orders separately for easiness of packing purpose for take-away customers. It's easy for serving staff to arrange the orders in frequent manner and that will help to serve customers as first come first out.

Cashiers

Currently they have to remember all the item codes, if there is a new employee it is very difficult to bill the order in cash register and its time consuming. If there is a method to see the items in category wise that would be easy and efficient.

By observing the working environment, it is very clear about the drawbacks of the current system. Current method causes customer dissatisfaction and due to that competitors also getting advantages.

3.2.1 Nature of the current method

The current method is based on papers. Only way to track order is cash register and the bill that given to the customer. Menu cards are also paper based or verbal. Those menu cards are printed and if any change is required restaurant must redesign the menu cards and reprint them. This is a big waste and it is costly and it is impossible to reprint every time even for a minor change. Current system is time consuming. Customers have to waste their time in queue and place the order. Several case of serving a wrong order are also happened due to miscommunication. Therefore, briefly,

For placing any orders customers have to visit restaurant to know about food items and then place order and pay. In this method time and manual work is required.

It is difficult to ensure whether the order has placed correctly or not because cash register codes are unknown to the customer. Sometime there are miscommunication between kitchen and front desk. No database is present; therefore, analysis is impossible [5].

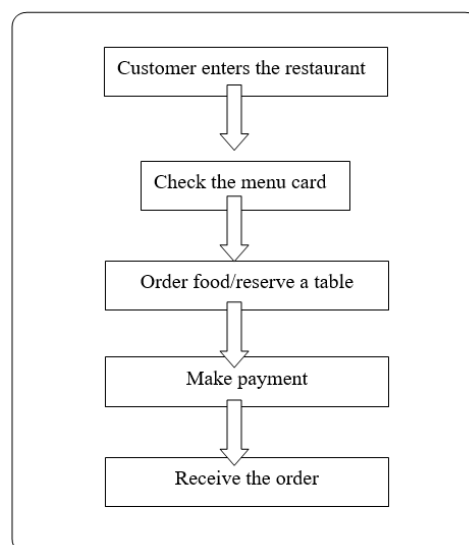


Figure 3.1: Current Restaurant Management

Figure 1.28 shows how current system works in ordered way. Other than cash register all the other processes are manual and paper based.

Analyzing current system methodologies will help to identify main system functionalities. Fact gathering techniques mainly observations and Interviews was carried out as domain analysis. Currently there is no computer use and cash register and paper work is using mostly for everything in this system. Other than that they do not use any proper method to measure food item demand or forecasting, stock reordering and moving patterns.

The most general modelling language to describe both the structure and behavior of a software system is Unified Modelling language (UML). Use case diagrams have already been used in the requirements analysis as a way to graphically overview the order process within the system. Other diagrams from the UML family are used in the design stage to show the structure and behavior of numerous sophisticated design features.

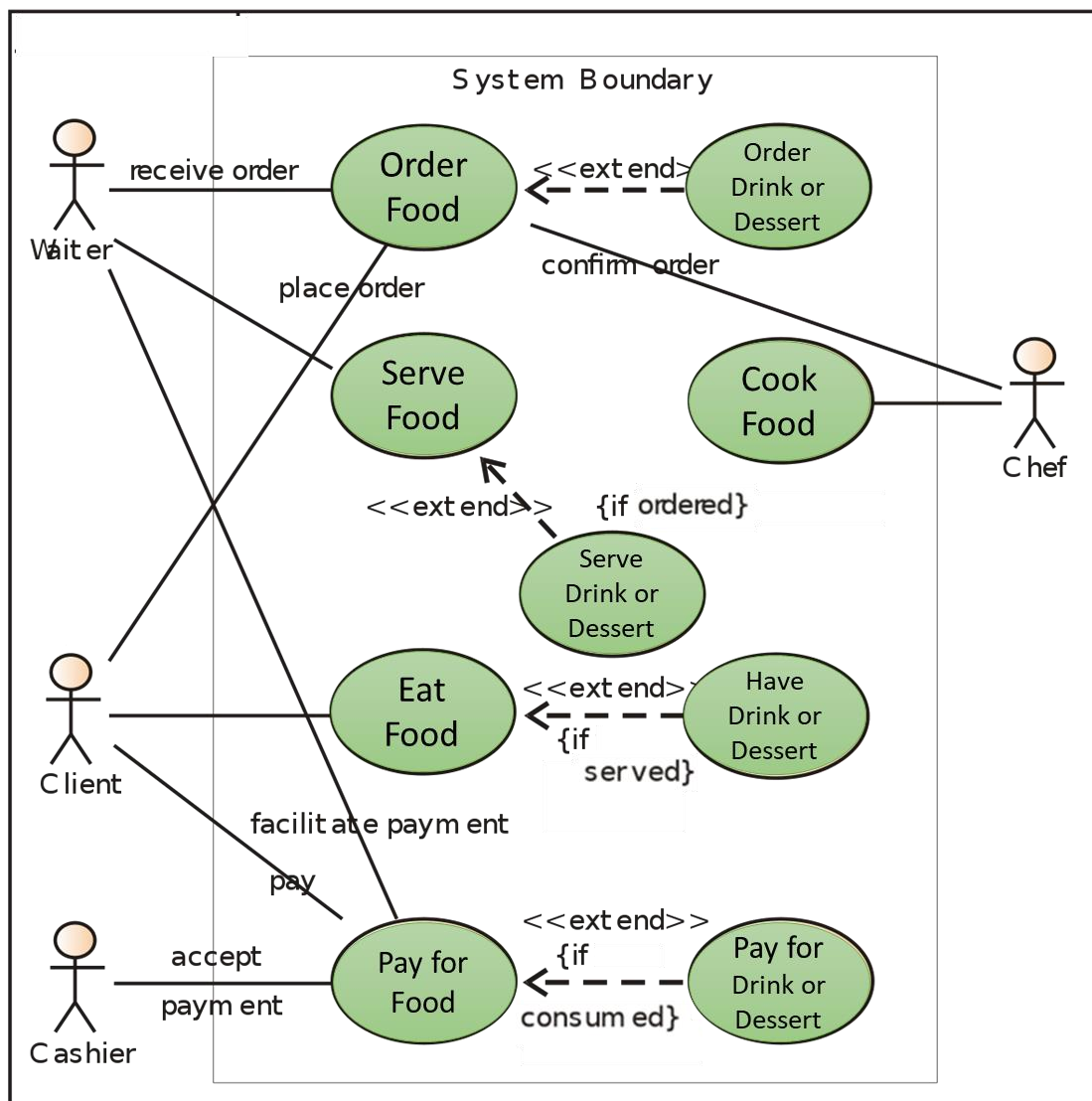


Figure 3.2 : High level Use case diagram representing the existing system.

3.2.2 Requirements of the proposed system

Web Ordering Function

This is the front end of the system. Customers can place their order and supply necessary details throughout this module. Following functionalities must be provided to the restaurant stakeholders.

Table 3-1: Required functionalities to the stakeholders

Feature	Customer	Cashier	Admin User	Chef
Account creation	✓	✓	✓	
Managing own account	✓		✓	
Log-In to the system	✓	✓		
Check Menu	✓	✓		
Select food Items	✓	✓		
Order placement	✓	✓		
Review/Update order	✓	✓		
View order	✓	✓		✓
Cancel order	✓	✓		
Make payment	✓	✓		
Table Reservation	✓	✓		

Menu Management Function

Provides functionality for the Admin User only. It will not be available to any other users of the system like Restaurant Employees or Customers.

Using a graphical interface, it will allow an Admin to manage the menu that is displayed to users of the web ordering system:

- Add/update/delete food category to/from the menu.
- Add /update/delete food item to/from the menu.
- Update price for a given food item.
- Update additional information (description, photo, etc.) for a given food item.

Before customers can actually use this system, functionality provided by this component will

have to be configured first. Once the initial configuration is done, this will be the least likely used component as menu updates are mostly seasonal and do not occur frequently [5].

Order Retrieval Function

Designed to be used by Admin and cashiers This function provides the following functions:

- New order retrieval from the database.
- Display the orders in an easily readable, graphical manner.
- Cashiers can view/Edit/Remove and make orders from visiting customers

Kitchen Function

This is the Simplest Module. Kitchen Staff and serving staff can use this. It provides flowing functions.

- View the orders as bulks for the easiness of preparation.
- Mark orders as ready
- Mark some items as Not available
- View orders in the sequence they come

Inventory Control Function

Only useable for the Admin User

- Update Stock Levels
- Inform when stocks are in re-order level

User Management function

Only useable for the Admin User

- Admin user can add/edit/remove cashier accounts
- Remove or edit customers

Reporting function

Only useable for the Admin User

- Generate useful reports

Discount and offers function

Only useable for the Admin User

- Add Discounts
- Edit Discounts

3.2.3 Non-functional requirements

Non-functional requirements do not directly affect the system. But rather than specific behaviors, these requirements are specific criteria that can be used to evaluate the working behavior of the system. If the system is failed to meet non-functional requirements sometimes whole system may unusable. Some constraints and restrictions can be considered as non-functional requirements when developing this kind of software. Quality behaviors, quality attributes and quality of service may fall in to this category [6].

Following are the non-functional requirements for this restaurant management system.

- The system should provide a user friendly environment including flexible interfaces,
- Person with average computer skills can work with the system with a short period of training.
- The system should be accurate and consistent, when manipulating the fed data in proper way and displaying correct information,
- The system should keep up security and reliability, because the system handles important data related to business processes of the company.
- Occasionally backups should be taken to maintain reliability The system should be reusable and maintainable.

3.2.4 Proposed architecture for the system

MVC or Model View Controller architecture was used to develop this system. As a software design pattern for developing web applications MVC is popular.

MVC architecture divides web application in to three parts. All those parts are interconnected. It is fully capable to support rapid web application development and dynamic interactivity with the database.

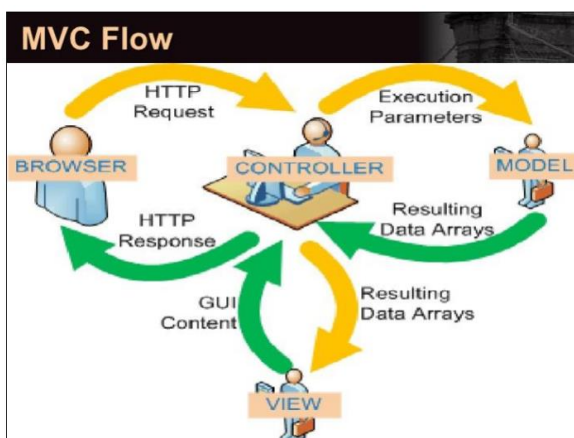


Figure 3.3: MVC framework

“Model

The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data. For example, a Customer object will retrieve the customer information from the database, manipulate it and update it data back to the database or use it to render data.” [7]

“View

The View component is used for all the UI logic of the application. For example, the Customer view will include all the UI components such as text boxes, dropdowns, etc. that the final user interacts with.” [7]

“Controller

Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. For example, the Customer controller will handle all the interactions and inputs from the Customer View and update the database using the Customer Model. The same controller will be used to view the Customer data.” [7]

3.3 Design

3.3.1 Design Techniques

To model the system, design techniques are used. Object Oriented design, Rapid application development, prototyping and Modern structured design are examples to design techniques that currently used.

For this development Object oriented design techniques has been used. In this method there are advantages such as, code reusable facility, design benefits and maintainable facilities with Objects and classes.

Unified Modelling Language (UML) plays a significant role in Object Orient designing. UML allows programmers easy understand models of objects so that programmers can easily write software. Some of structural and behavioral UML diagrams use for design proposes systems are mentioned below.

- Use case diagrams – This makes clear what are the capabilities of the system.
- Use case narratives – Make use cases more clear

- Activity diagrams – shows how activities are coordinated to provide a service which can be at different levels of abstraction.
- DFD's – Data flow diagrams representing a flow of data through a process or a system
- Data flow Diagrams – shows how the objects interact overtime.
- Class diagrams – this is the main building block in object oriented modelling [8].

Use case diagrams

Overall high level Use-Case is shown in this figure 3.4.

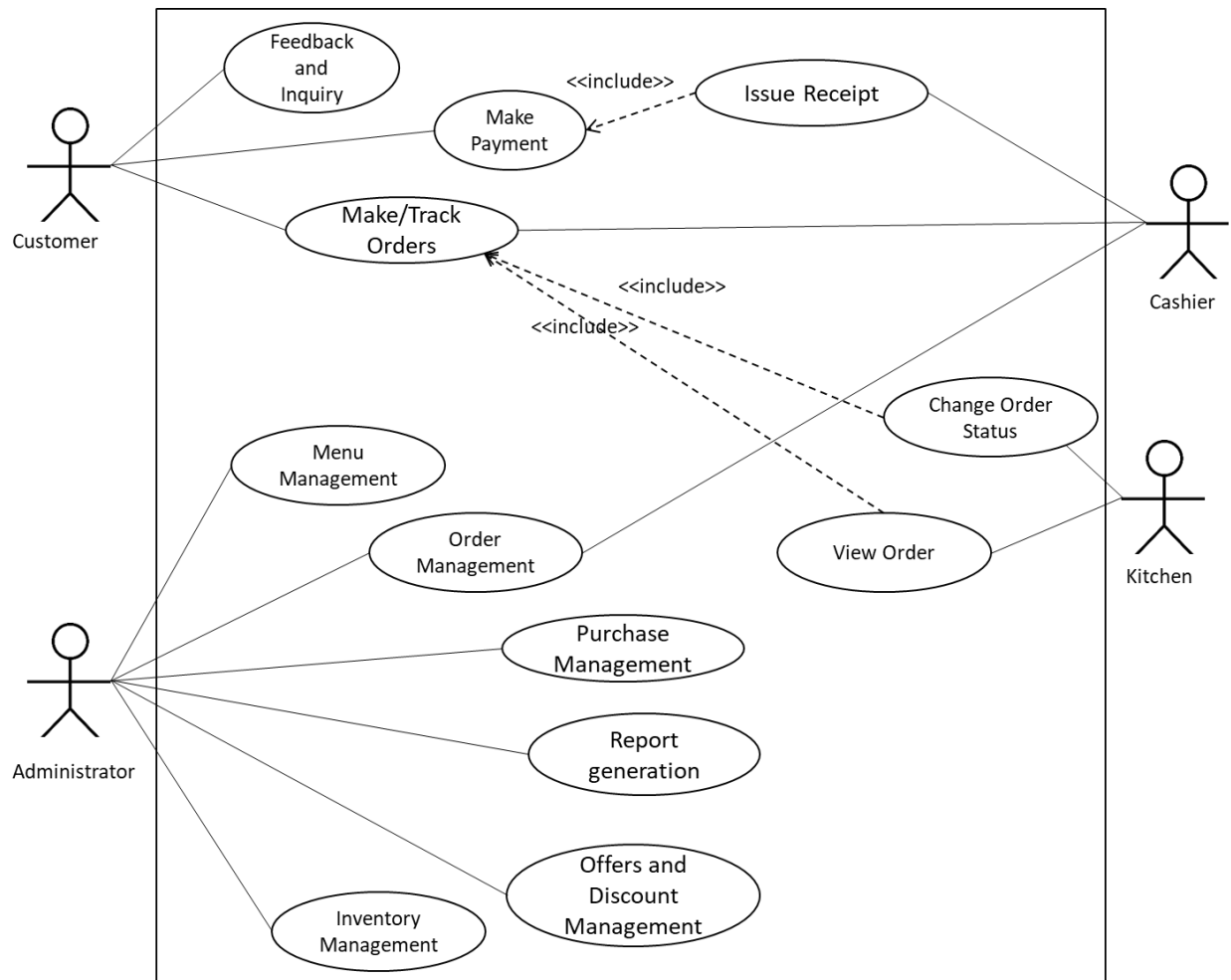


Figure 3.4: Overall High-level Use case for the proposed system

Class Diagram

Class diagram is the main building box of object-oriented modelling. General conceptual modeling of the structure of the application, and for detailed modeling translating the models into programming code was done using this.

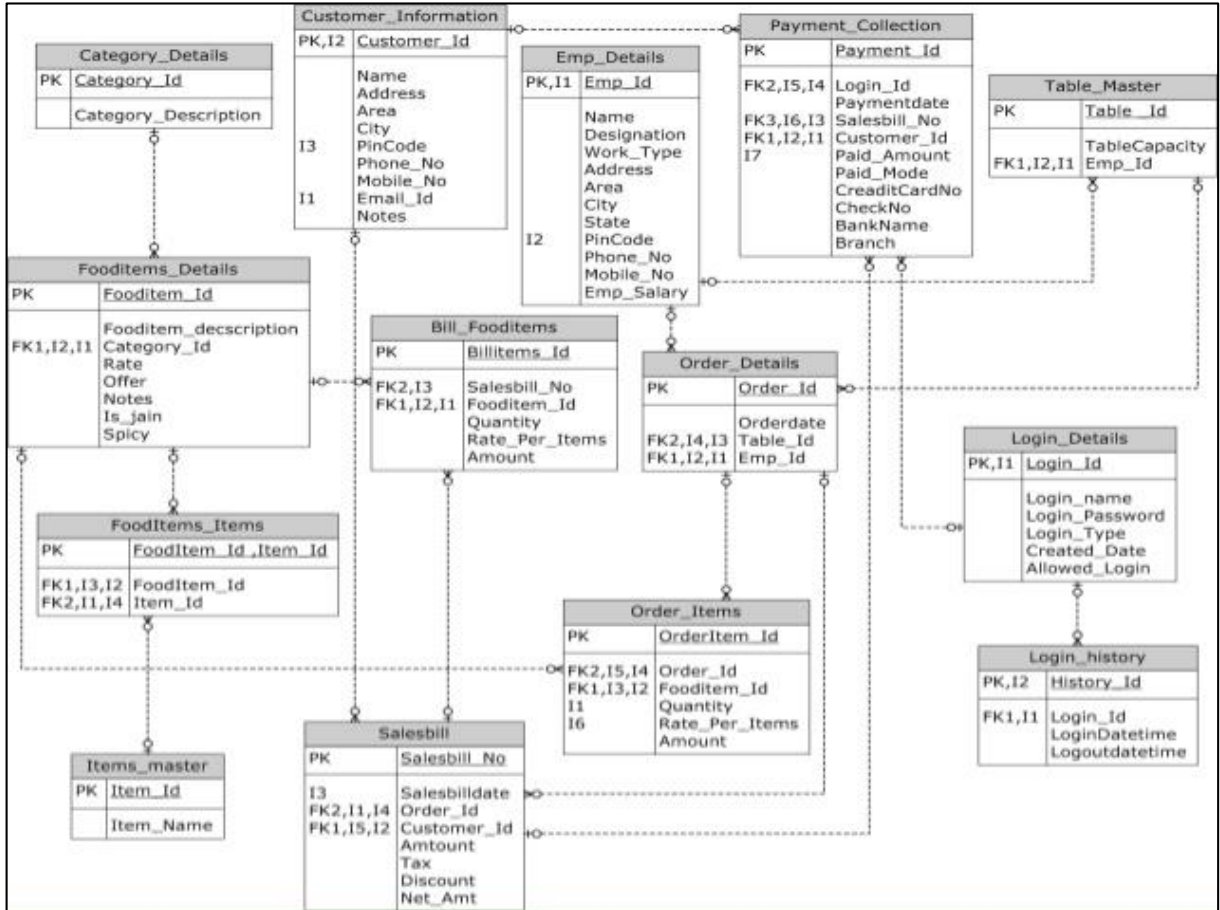


Figure 3.5: Class diagram

Activity Diagram

“Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

Activity diagram is suitable for modeling the activity flow of the system. An application can have multiple systems. Activity diagram also captures these systems and describes the flow from one system to another. This specific usage is not available in other diagrams. These systems can be database, external queues, or any other system” [8].

Following figure 3.6 shows activities to order food online.

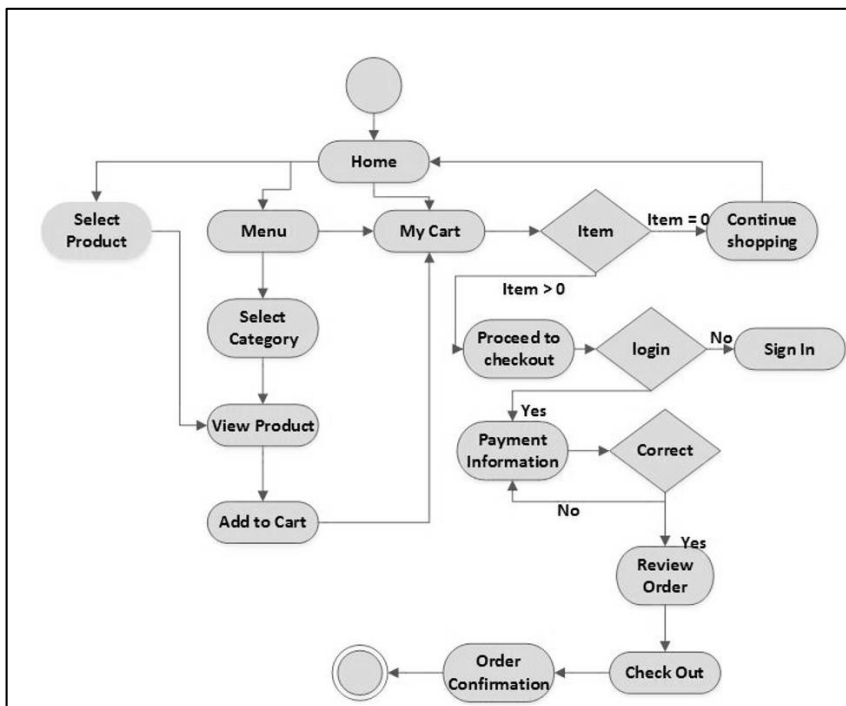


Figure 3.6: Activity diagram- Customer order

Data Flow Diagram

Data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops.

Refer appendix for Level 0, Level 1 and level 2 diagrams.

3.3.2 User Interface Design

User interface is the primary and very first interaction with user and system. So it should be interactive to user. User Interface (UI) Design focuses on what users expect through the system and ensuring that the interface has fundamental facilities which are easy to understand, access, and use. There are some properties that a user interface must have,

- Avoid unnecessary elements and keep the interface simple:
- Purposefully use color and texture: make direct attention toward or redirect attention away from items using color, texture, contrast.
- Use auto-select options as well as let the users select options for better flexibility.
- Use common UI elements and build consistency: By using more comfortable common elements, for users and able to get things done fast.
- Use typography to create hierarchy and clarity: Different sizes, fonts, and arrangement of the text to help increase scalability, legibility, and readability.

3.3.3 User Privileges.

This figure 3.7 shows the privileges users have for this system.

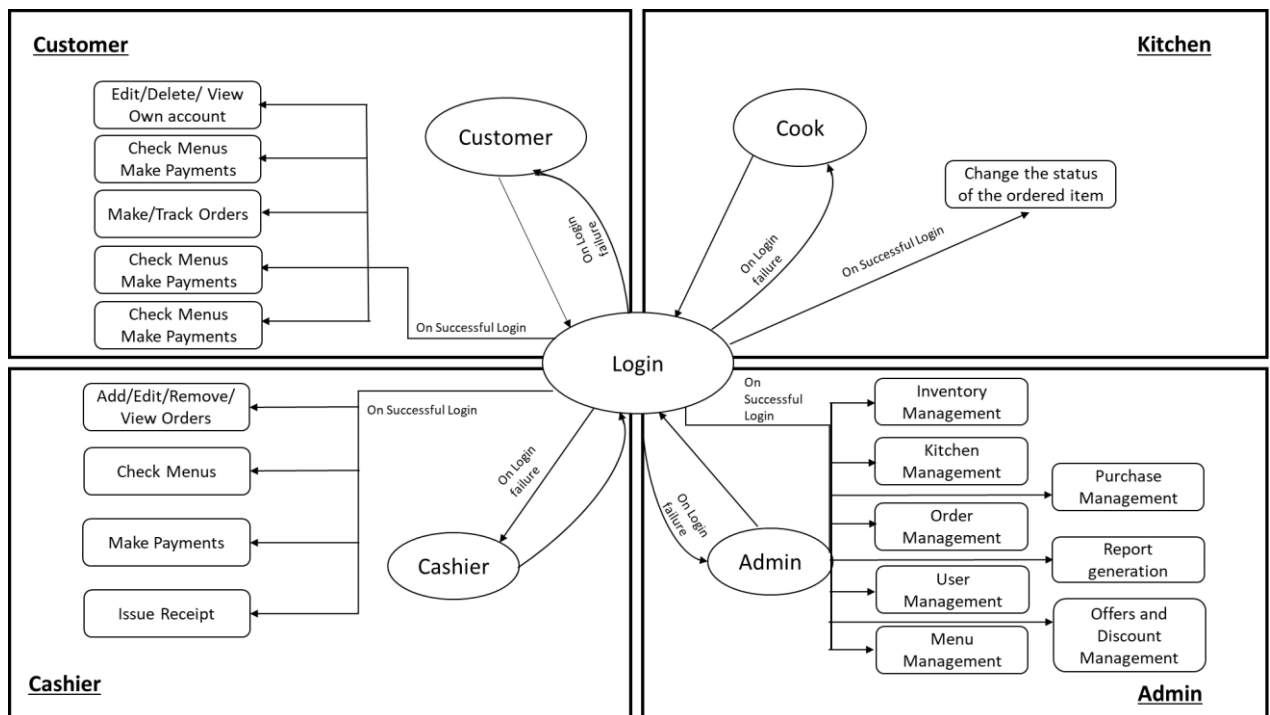


Figure 3.7: User Privileges

3.4 Summary

Discussed about the requirements and designing of the system. Implementation of the system is discussed in next chapter.

Chapter 4 Implementation

4.1 Introduction

Implementation process use to convert system specifications to executable system. In this phase convert design to user friendly views with font end interfaces and back end systematic logics for perform functionalities. A comfortable language and suitable tools were selected in the process of coding and development. The codes were written and arranged in a readable format. Comments added for special logics to help for future implementations.

4.2 Implementation Environment

In Implementation process both server side and client environments were considered. There are environment requirements in this process. Those requirements are divided as software and hardware requirements.

4.2.1 Operating Environment

This requires active internet connection because this is web based.

- Minimum Windows 8.1 Operating system - 32 or 64 Bit. (earlier Windows 7 was enough, but due to discontinuation of updates from Microsoft windows this was changed.)
- 2 Computers (One for the cashier and one the admin user)
- 2 android Tabs (for chefs)
- Active Internet Connection

4.3 Development tools and techniques

4.3.1 Tools

- PhpStorm - PhpStorm is a cross-platform IDE for PHP. PhpStorm provides an editor for PHP, HTML and JavaScript with on-the-fly code analysis, error prevention and automated refactoring for PHP and JavaScript code.
- MySQL - all database related developments were handled using this because development tools, and administration objects are available with this software.
- Adobe Photoshop CC 2018 was used to edit photos

4.3.2 Technologies

- HTML – Hyper Text Markup Language is used to build the base interfaces of the system
- CSS – Cascading Style Sheets was used to make the system more attractive and user friendly.
- Php –used to build the system
- JavaScript- this is a programming tool to web development and used for validations.
- JavaScript based AJAX – Used to get data from server. Use of this avoid the requirement of refreshing it continuously.
- JQuery - Which is also based on JavaScript was used to implement some features.
- Lanka Hosting Space- Paid service used here to host the website.

4.4 Reusable Components

- Date picker – To select a date from the calendar.
- Data table – Table adept of searching and filtering data, pagination of data.
- CSS – Cascading Style Sheet to design an interactive user interface.
- Web Template –Front end Design which can edit
- Charts - Use to generate HTML based JavaScript based interactive charts to display information graphically to user.

4.5 Network Architecture

Users can interact with the restaurant management system using web browser. Can be access via Internet and this system has central web server, for users to connect using their internal network in future.

4.6 Application Development Architecture

As discussed in the previous chapters, the system is built on top of MVC. The Model is used for database interaction. The Controller is to send requests and responses back and forth between models, views. Graphical user interfaces represent by Views

4.7 Main User Interfaces

Only a few number of user interfaces and wireframes are provided in this section to show the structure of the system. Please Refer Appendix C - User Documentation for the rest of the interface designs.

4.7.1 Admin Dashboard

Controlling the system can be done using this panel. Administrator have privileges to do following functions

1. Inventory Management
2. Menu Management
3. User Management
4. Order Management
5. Purchase Management
6. Offers and Discounts
7. Report Generation
8. Kitchen Management

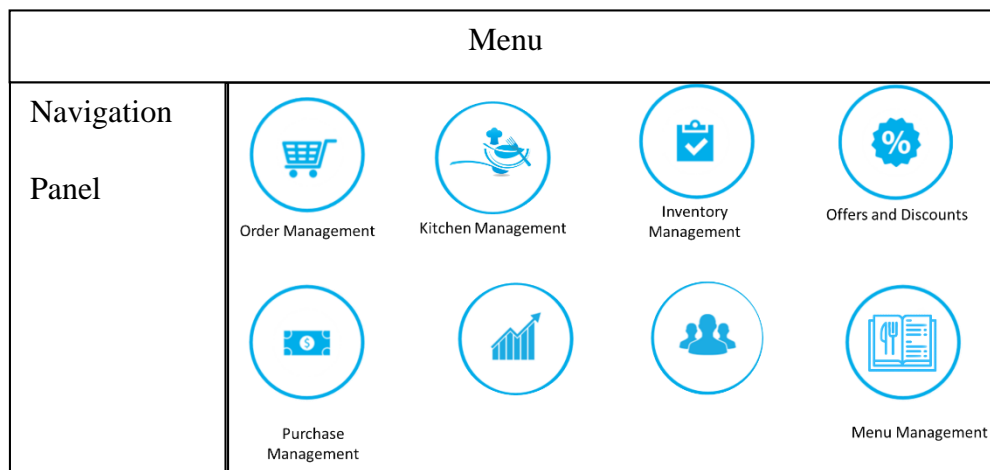


Figure 4.1: Admin Dashboard

4.7.2 Order Food Cashier

Cashier can easily order food through this interface.

Customer Name/Unregistered Customer		Customer Number
Order Type	Item List	
Cashier Name	Item 1	Item 2
Current Order Details		
		Total

Figure 4.2: Cashier Order Module

4.7.3 Kitchen Module

Kitchen staff can view orders as they arrived. They can view orders as bulks. Kitchen staff can change the status of the order

Kitchen Order Ticket	
ITEM 1	Total
	Order 1
	Order 2
	Order 3
ITEM	Total

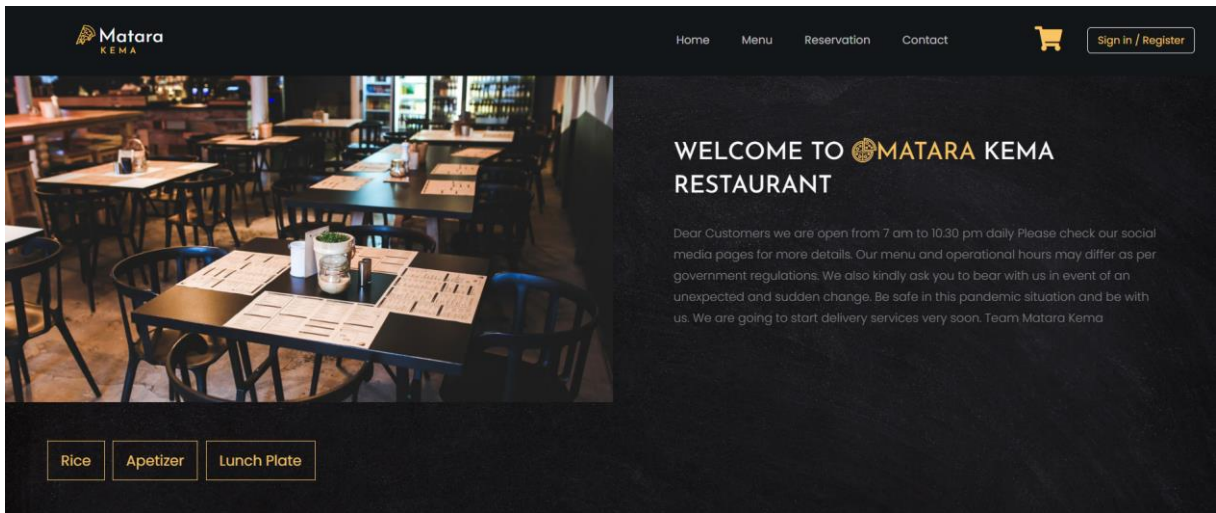


Figure 4.4: Home Page of the system

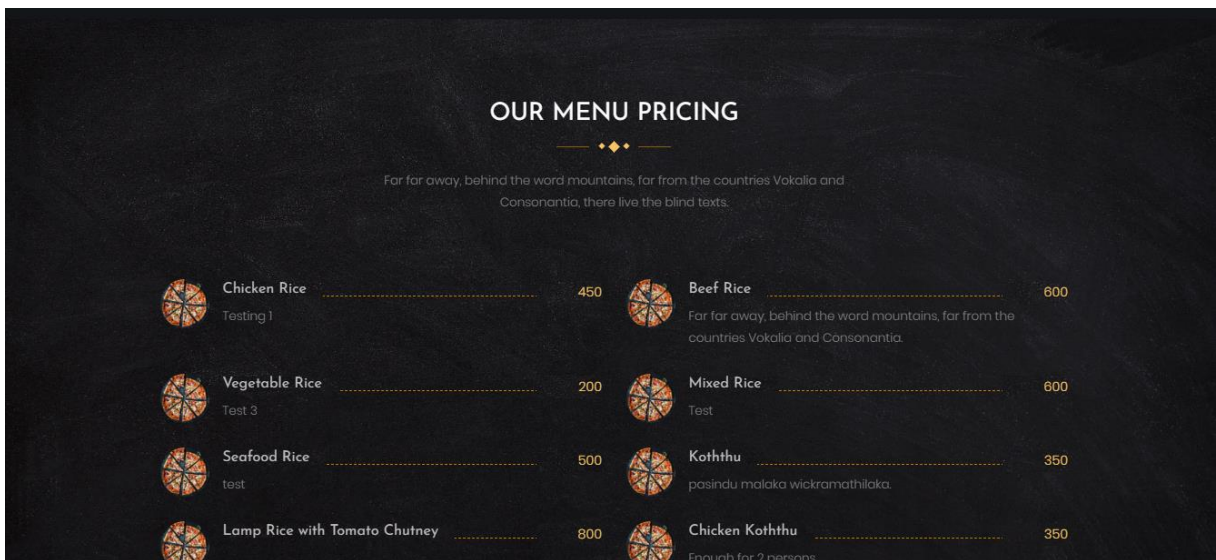


Figure 4.5: Food Menu

4.8 Summary

This chapter has discussed the interesting aspects from the implementation stage. The next chapter documents the results by demonstrating the working system

Chapter 5 User Evaluation and Testing

5.1 Introduction

Software testing is a critical component of software quality assurance that represents the ultimate analysis of specification, design, and code generation of software product. The testing method is basically combine with Verification and Validation. Validation refers to testing whether the system satisfies the requirements while verification refers to whether the system implements the specified functions properly. Basic goals of test evaluation is determining whether the promises about the invention by the supplier and the requirements of the customer are met on an acceptable level.

5.2 Testing Procedure

Software testing and implementation are iterative processes. Most of the time both stages work as simultaneous system components. The propose system was tested while the development was ongoing. Following iterative RUP development life cycle made it possible to test Iterative increments of the software.

Structural testing techniques were followed in this phase including “white box” testing which tests “how a program/system does something”. Functional testing techniques were used, which includes “black box” testing which tests the behavior of a system or program. These techniques were exploited in different testing levels like unit testing.

System followed object oriented methodology, so object oriented testing was also carried out in this phase. Individual operations associated with objects were tested initially, followed by testing individual classes and clusters of objects, and finally test the system as whole. User acceptance testing was completed in client site, participation of few staff members.

5.3 Test Plan and Test Cases

Implementation stage starts the testing process. In development stage code is reviewed. Test plan included all phases of testing and also used as a guide for the overall testing process. The test plan was designed before the implementation of the system. Test Objectives, Test Schedule test Logistics and Test Strategies are included in Test plan. Test cases are highly considered in here.

Test cases were created according to the designed test plan. That contains data, procedure, and expected result and represents which use to system or part of the system run. To reduce complexity of the testing process test cases were designed for each module independently. The following tables specify some test cases.

Manual testing method and procedures used for testing rather than automation tools and technologies.

Below tables tabulates some test cases for test basic functions.

5.3.1 Master Data modules

These tables tabulate some of the test cases for testing basic functions.

Table 5-1: Add customer test case

Test Case ID	1		
Tested Component	Add Customer		
Module Name	Customer Module		
Test Case	Add new customer		
Expected output	Add customer successfully, View should be user friendly. If adding customer form missing any required information error message must be displayed.		
Test Case Description			
No	Test Case	Actual Output	Status
01	(Positive) Add customer success	Enter customer information (Title, First Name, Last Name, Street, Town, Phone No, Email, Remarks) (Title, First Name, Last Name, Street, Town, Phone No, Email, must be filled save customer) Validate Then redirect to relevant home page (Admin – Add customer)	Pass
02	Negative Log-In Not success	Enter customer information (Title, First Name, Last Name, Street, Town, Phone No, Email, Remarks) (Not valid) Text box show error Message- “Required field”	Pass

5.4 Acceptance Testing

Actual working environment of Matara Kema Restaurant was used for acceptance Testing. Real Transaction data sets were used and users were restaurant staff. Employees were asked to work with related modules in system according to their privileges. Activities were monitored while users working with the system.

Final outcome must be handed order to actual end users for their acceptance. Due to Covid- 19 problem, still it is not practical to meet real end users in actual situation.

5.5 Summary

This chapter explains about full testing cycle confirmed that developed system is functioning successfully. Next chapter will give the conclusion.

Chapter 6 Conclusion

6.1 Introduction

This is the final chapter of this dissertation. Objectives achieved, lessons learnt, Problems and future enhancements will be discussed here. Matara Kema restaurant is become more popular than previous times. They have a plan to start delivery services also. Therefore, it has been implemented as a web based system to meet the requirements of increasing demand

This system helps to manage entire restaurant easily. Restaurant operations also can be well manage using this solution.

6.2 Problems Encountered

The client has no any computer knowledge and due to that it was difficult to capture the requirement. Usually staff who works in this kind of place also less knowledgeable and they are afraid to work with computers. It was really difficult to explain about this system, and gather the requirements they need.

Due to medical condition, I had to omit some parts of the system because I was not able to work more than one month.

Covid-19 situation also made so many difficulties to development of this system because agile methodology is used. Testing was not able to done properly because even after 3 months still restaurant is not operating properly.

Lack of knowledge about development tools and hosting environment was also a huge problem. Books, Video lessons and support from knowledgeable people were used to overcome this problem to some extent.

6.3 Lessons Learnt

Throughout this project, valuable knowledge has been gained. This is the first experience to work with a client that has no IT knowledge. Learned many things about fact gathering techniques in actual environment. This project gave me the opportunity to gain experience of actual web developer. It helps to understand how project planning should be done and had opportunity to collaborate with many Individuals and improve my skills. Had a great lesson about time management also.

This project gave chance to gain knowledge about development tools such as phpStrome, my SQL ect. This was the first experience about web hosting and domains.

Theories techniques and technologies that learnt throughout the MIT degree program was very valuable for this project.

6.4 Future enhancements

Delivery function has to added to the system because restaurant in planning to give that facility.

Most of the customers suggested about mobile application and planning to implement that function also.

Report generation part must be implemented more than this.

References

- [1] "How to Keep Restaurant Customers Coming Back for More | Smart Web Restaurant", *Swebrestaurant.com*, 2019. [Online]. Available: <https://swebrestaurant.com/blog/article/keeping-customers-coming-for-more>. [Accessed: 19-Jan- 2019].
- [2] "Restaurant Management System Small Business Guide - business.com", *business.com*, 2020. [Online]. Available: <https://www.business.com/articles/restaurant-management-system-guide/>. [Accessed: 20- Jan- 2020].
- [3] "How to Keep Restaurant Customers Coming Back for More | Smart Web Restaurant", *Swebrestaurant.com*, 2019. [Online]. Available: <https://swebrestaurant.com/blog/article/keeping-customers-coming-for-more>. [Accessed: 19-Jan- 2019].
- [4] M. Chauhan, "How Restaurant Inventory Management System Simplifies Tracking and Running? - Asset Infinity", *Asset Infinity Blog*, 2020. [Online]. Available: <https://www.assetinfinity.com/blog/restaurant-inventory-management-system-simplifies-tracking>. [Accessed: 11- Feb- 2019].
- [5] M. Patel, "Online Food Order System for Restaurants", 2015. Available: <https://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1222&context=cistechlib>. [Accessed 20 February 2020].
- [6] "Non-functional requirement", *En.wikipedia.org*, 2020. [Online]. Available: https://en.wikipedia.org/wiki/Non-functional_requirement. [Accessed: 21- Mar- 2020].
- [7] "MVC Framework - Introduction - Tutorialspoint", *Tutorialspoint.com*, 2020. [Online]. Available:https://www.tutorialspoint.com/mvc_framework/mvc_framework_introductionht

Appendix A System Design

Use Case Diagram

Customer Order Food Online

Use case	Customer Order food online
Actors	Customer, Payment gateway
Overview	
Customer order food online	
Pre-conditions	
Customer have to register on the system to order food online	
Flow of events	
Customer check menu	
Select item	
Place order	
Confirm order	
Make payment	
Post conditions	
Order details updated to system, Order view in Kitchen display	

Table A.1 Customer Order Food Online

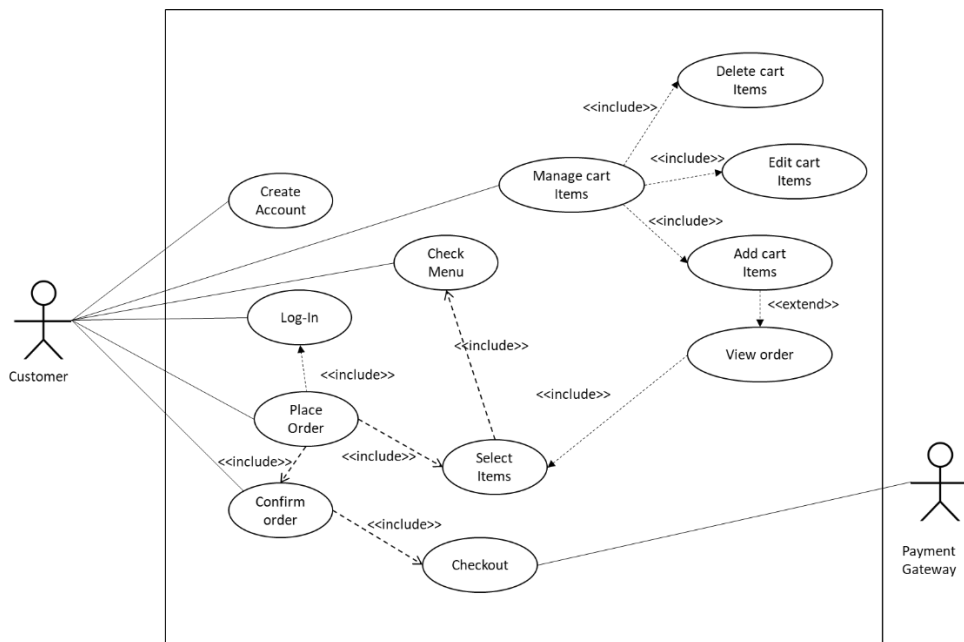


Figure A.1: Customer Module

Data flow diagrams

Level 0 diagram

Entity	Description
customer	Use to provide food order
Admin	Define menu details
Kitchen	View orders
Cashier	Get order details, Generate receipt

Table A.2 : Level 0 diagram

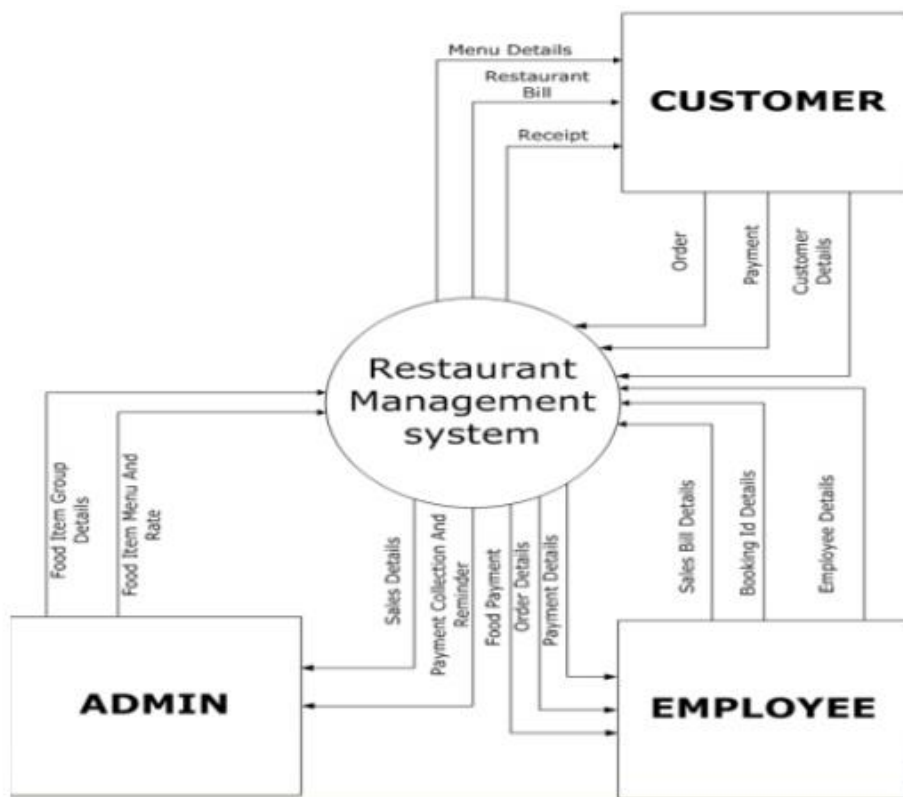


Figure A.2 level diagram

Level 1 Diagram

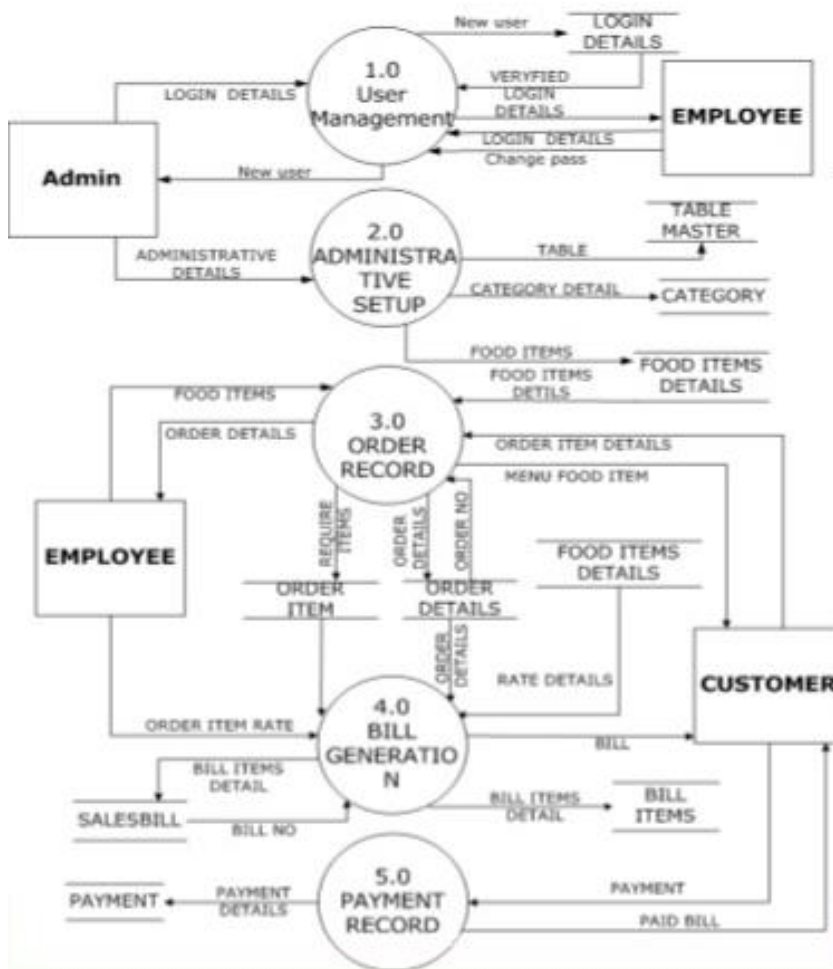


Figure A.3 : level 1 diagram

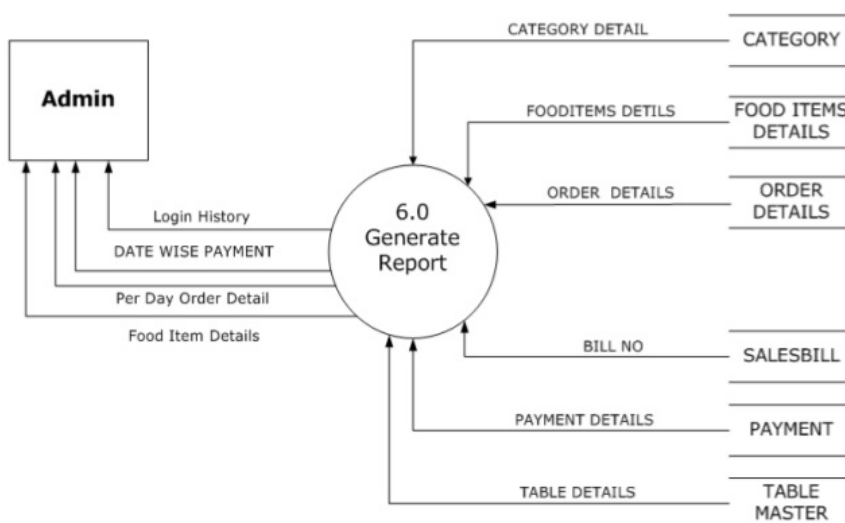


Figure A.4: Level 2 diagram

Level 2 Diagram

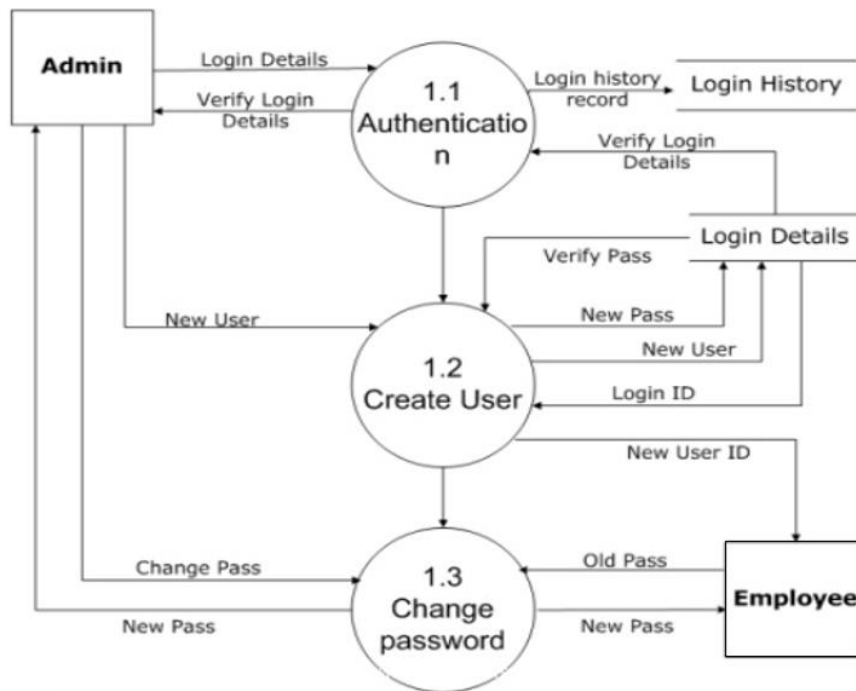


Figure A.5: Level 2 diagram

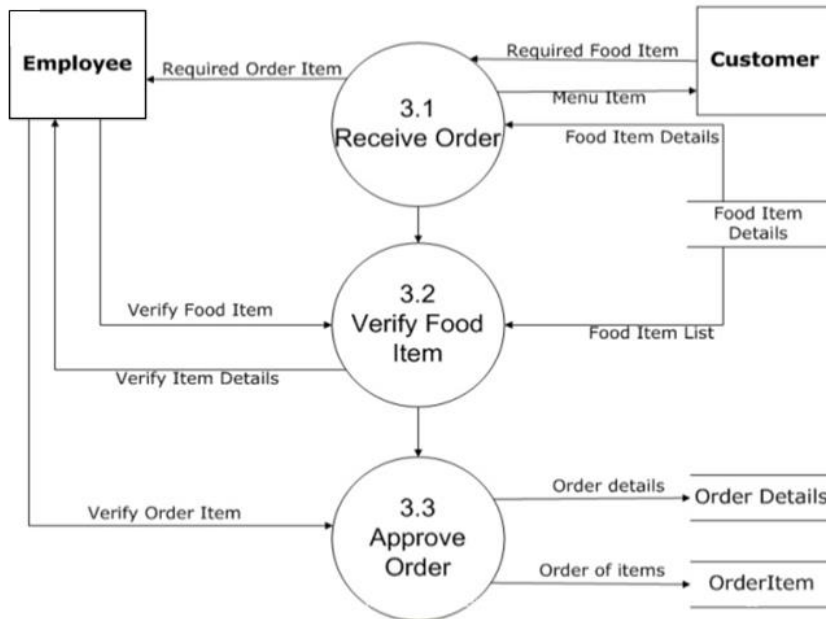


Figure A.6: Level 2 diagram-2

Data Dictionary

Category Details

Column Name	Data Type	Size	Description	Constraint
Category_Id	Numeric	(2, 0)	Id of the category.	Primery key
Category_Description	Varchar	(20)	Foods of category	Not null

Table A.3 : Category Details

FoodItems_Details

Column Name	Data Type	Size	Description	Constraint
FoodItem_Id	numeric	(5, 0)	Id of the FoodItem.	Primary Key
FoodItem_Name	Varchar	(50)	Name of FoodItem.	Not null
Category_Id	numeric	(5,0)	Id of the Category.	Not null
Rate	numeric	(2)	Rate of Item.	Not null
Offer	Varchar	(20)	Offer	Not null
Notes	Varchar	(50)	Any Notes	
Is_jain	Yes/No		FoodItem is Isjain or not	
Spicy	Varchar	(6)	High / Medium / Simple	Not null

Table A.4 : FoodItems Details

Item_Master Table

Column Name	Data Type	Size	Description	Constraint
Item_Id	numeric	(3,0)	Id of the Item.	Primery key
Item_Name	Varchar	(20)	One fooditem contain many items	Not null

Table A.5: Item Master Table

Table_Master

Column Name	Data Type	Size	Description	Constraint
Table_Id	numeric	(2,0)	Id of the Table Master	Primary key
TableCapacity	Varchar	(30)	Capecity of customer in one table	Not null
Emp_Id	numeric	(2,0)	Id of the Employee	Foreign key

Table A.6: Table Master

Order_Details

Column Name	Data Type	Size	Description	Constraint
Order_Id	Numeric	(5,0)	Id of the Order.	Primary key
Orderdate	Datetime	-	Date of the Order	Not null
Table_Id	Numeric	(2,0)	Id of the Table Master	Foreign key
Emp_Id	Numeric	(2,0)	Id of the Employee	Foreign key

Table A.7: Order Details Table

User Interfaces

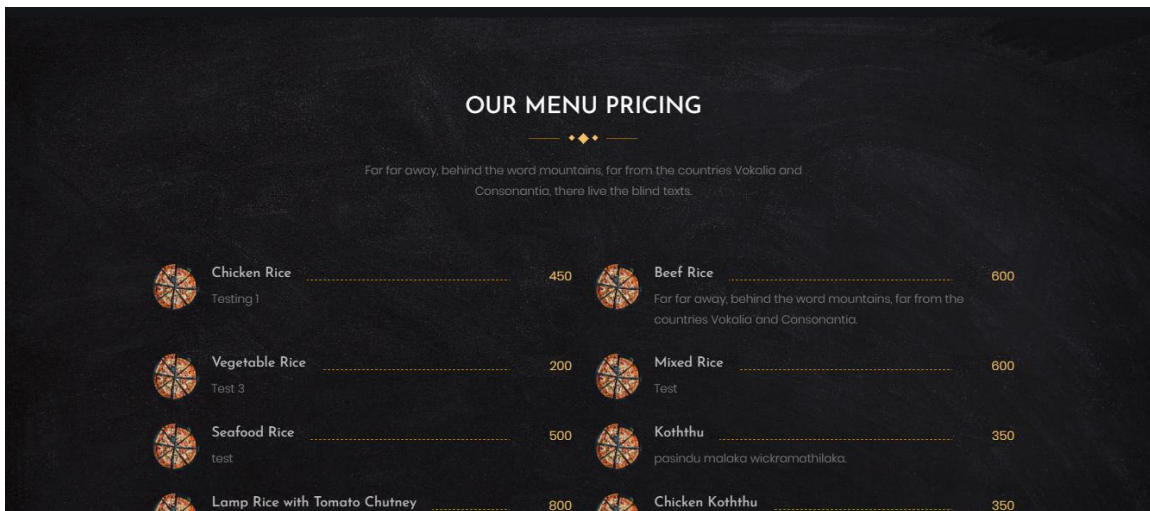


Figure A.7 : Menu Pricing

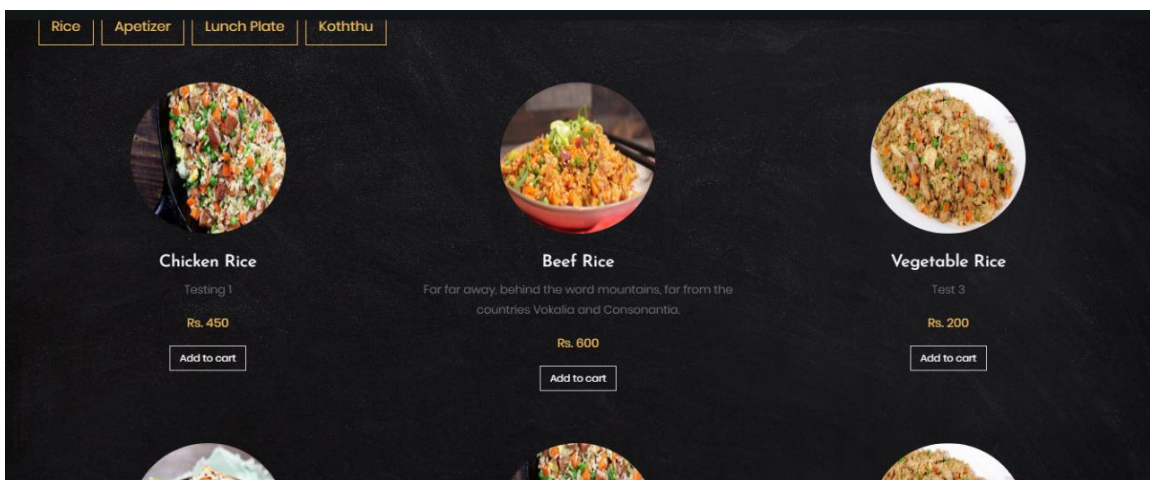


Figure A.8: Sub Menu

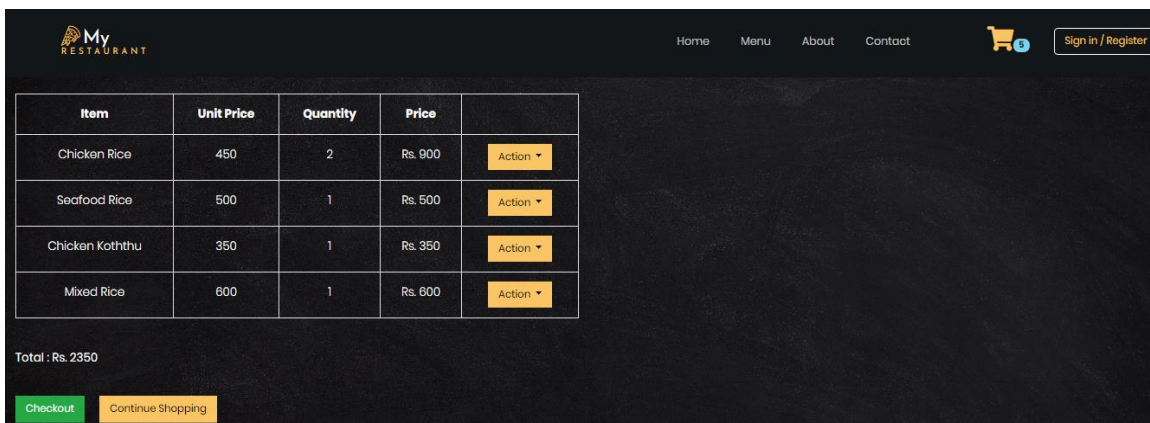


Figure A.9: Selecting food item